

PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION  
International Bureau



INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification <sup>6</sup> : <b>H02P</b>		A2	(11) International Publication Number: <b>WO 97/15111</b> (43) International Publication Date: 24 April 1997 (24.04.97)
(21) International Application Number: PCT/US96/16728		(81) Designated States: AL, AM, AT, AU, AZ, BB, BG, BR, BY, CA, CH, CN, CZ, DE, DK, EE, ES, FI, GB, GE, HU, IS, JP, KE, KG, KP, KR, KZ, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, TJ, TM, TR, TT, UA, UG, US, UZ, VN, ARIPO patent (KE, LS, MW, SD, SZ, UG), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).	
(22) International Filing Date: 17 October 1996 (17.10.96)			
(30) Priority Data: 60/005,677 17 October 1995 (17.10.95) US			
(71) Applicant (for all designated States except US): FASCO INDUSTRIES, INC. [US/US]; Suite 200, 500 Chesterfield Center, Chesterfield, MO 63017 (US).			
(72) Inventors; and		Published	
(75) Inventors/Applicants (for US only): GUNNELS, Ronald, Lee [US/US]; 7082 E. Farm Road, #164, Rogersville, MO 65807 (US). SNIDER, Jeffrey, R. [US/US]; 3668 W. Maplewood, Springfield, MO 65807 (US). BUTTRAM, Darrel, C. [US/US]; Route 2, Box 217B, Crane, MO 65633 (US).		Without international search report and to be republished upon receipt of that report.	
(74) Agents: PERREAUXT, Donald, J. et al.; Lorusso & Loud, 440 Commercial Street, Boston, MA 02109 (US).			
(54) Title: A BRUSHLESS DC MOTOR ASSEMBLY			
(57) Abstract			
<p>A brushless dc motor assembly including a brushless dc motor, and a control board having at least one output connected to a stator winding of said brushless dc motor for providing current flow to said stator winding. Control electronics on the control board control current flow to said stator winding using a microprocessor and based on desired motor operating characteristics. The control electronics include a housekeeping power supply for providing a stable 5v DC signal from a rectified AC line voltage. The control electronics also include a MOSFET output amplifier having a power zener diode connected to the drain thereof, said power zener dissipating temporary back emf resulting from switching of said MOSFET from an on to an off state. A hall device mounted to a stator of said brushless dc motor is also provided. The hall device provides a signal representative of the rotational speed of a rotor of said motor to the control board. The control electronics on said control board control said current flow to said stator winding responsive to said signal from the hall device. The control board is preferably attached to a heatsink whereby the heatsink is attached to MOSFETS on said control board for dissipating heat generated by said MOSFETS.</p>			

**FOR THE PURPOSES OF INFORMATION ONLY**

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AM	Armenia	GB	United Kingdom	MW	Malawi
AT	Austria	GE	Georgia	MX	Mexico
AU	Australia	GN	Guinea	NE	Niger
BB	Barbados	GR	Greece	NL	Netherlands
BE	Belgium	HU	Hungary	NO	Norway
BF	Burkina Faso	IE	Ireland	NZ	New Zealand
BG	Bulgaria	IT	Italy	PL	Poland
BJ	Benin	JP	Japan	PT	Portugal
BR	Brazil	KE	Kenya	RO	Romania
BY	Belarus	KG	Kyrgyzstan	RU	Russian Federation
CA	Canada	KP	Democratic People's Republic of Korea	SD	Sudan
CF	Central African Republic	KR	Republic of Korea	SE	Sweden
CG	Congo	KZ	Kazakhstan	SG	Singapore
CH	Switzerland	LI	Liechtenstein	SI	Slovenia
CI	Côte d'Ivoire	LK	Sri Lanka	SK	Slovakia
CM	Cameroon	LR	Liberia	SN	Senegal
CN	China	LT	Lithuania	SZ	Swaziland
CS	Czechoslovakia	LU	Luxembourg	TD	Chad
CZ	Czech Republic	LV	Latvia	TG	Togo
DE	Germany	MC	Monaco	TJ	Tajikistan
DK	Denmark	MD	Republic of Moldova	TT	Trinidad and Tobago
EE	Estonia	MG	Madagascar	UA	Ukraine
ES	Spain	ML	Mali	UG	Uganda
FI	Finland	MN	Mongolia	US	United States of America
FR	France	MR	Mauritania	UZ	Uzbekistan
GA	Gabon			VN	Viet Nam

1                   **A BRUSHLESS DC MOTOR ASSEMBLY**

2

3                   **Field of the Invention:**

4                  The present invention relates to a brushless dc motor  
5                  assembly, and more particularly, to a brushless dc motor which is  
6                  capable of receiving AC input from a wall outlet or other source.

7

8                   **Background of the Invention:**

9                  Brushless DC motors are widely used in due to their simplicity  
10                 of design, and high efficiency. Difficulty has been encountered,  
11                 however, in adapting brushless dc motors for receiving an AC input.  
12                 To do this, the common approach has been to provide the motor with  
13                 an externally mounted rectifier circuitry, typically including a  
14                 step-down transformer. This construction, however, causes a  
15                 significant increase in the overall size of the device which, of  
16                 course, is undesirable in many applications. In addition, wide  
17                 variations in AC input voltage to rectifier circuitry results in  
18                 wide variations in the DC input to the motor, thus affecting motor  
19                 performance and controllability. Finally, existing brushless dc  
20                 motors are only capable of operating under relatively low power,  
21                 e.g. 5 watts.

22                 In view of these and other disadvantages of the prior art,  
23                 there is a need in the art for a brushless dc motor which is  
24                 capable of being powered by an AC source (e.g. a common wall  
25                 outlet) and is efficient, compact, and cost effective.

1       **Objects of the Invention**

2           Accordingly, it is an object of the present invention to  
3       provide a brushless dc motor which is capable of operating from an  
4       AC input source.

5           Another object of other present invention is to provide a  
6       brushless dc motor which operates at high power levels.

7           Yet another object of the present invention is to provide a  
8       brushless dc motor which includes a control board for allowing  
9       customized control of motor parameters.

10          Still another object of the present invention is to provide a  
11       brushless dc motor having an internal house keeping power supply  
12       which is capable of providing a stable DC voltage to control  
13       electronics over a wide range of AC input voltages.

14          Still another object of the present invention is to provide a  
15       brushless dc motor having internal motor drive electronics  
16       including a power zenor diode for eliminating the detrimental  
17       effects of back EMF from the motor.

18          Still another object of the present invention is to provide a  
19       brushless dc motor having control and drive electronics including  
20       MOSFETS for increasing efficiency and a heat sink for dissipating  
21       heat generated by the MOSFETS.

22          Still another object of the present invention is to provide a  
23       brushless dc motor having a rotor shaft with a plastic or ceramic  
24       magnet and a bifiler wound stator.

1 Still another object of the present invention is to provide a  
2 brushless dc motor having a conduit box housing for the control  
3 electronics mounted on a heat sink.

4 These and other objects of the present invention will become  
5 apparent from a review of the description provided below.  
6

7 **Summary of the Invention:**

8 The present invention relates to a brushless dc motor  
9 assembly including a brushless dc motor, and a control board having  
10 at least one output connected to a stator winding of said brushless  
11 dc motor for providing current flow to said stator winding.

12 The control electronics on the control board control current  
13 flow to said stator winding based on desired motor operating  
14 characteristics. The control electronics include a housekeeping  
15 power supply for providing a stable 5v DC signal from a rectified  
16 AC line voltage, and a microprocessor for controlling output to the  
17 stator windings according to desired operating characteristics.  
18 The control electronics also include a MOSFET output amplifier  
19 having a power zener diode connected to the drain thereof, said  
20 power zenor dissipating temporary back emf resulting from switching  
21 of said MOSFET from an on to an off state.

22 A hall device mounted to a stator of said brushless dc motor  
23 is also provided. The hall device provides a signal representative  
24 of the rotational speed of a rotor of said motor to the control  
25 board. The control electronics on said control board control said  
26 current flow to said stator winding responsive to said signal from

1       the hall device. The control board is preferably attached to a  
2       heatsink whereby the heatsink is attached to MOSFETS on said  
3       control board for dissipating heat generated by said MOSFETS.  
4

5       **Brief Description of the Drawing:**

6       A preferred embodiment of the invention is described below  
7       with reference to the following figures wherein like numerals  
8       represent like parts:  
9

10      FIG. 1: is a front view of a motor assembly according to the  
11       present invention.

12

13      FIG . 2: is a side view of a preferred rotor assembly according to  
14       the invention.

15

16      FIG . 2a is an end view of the rotor assembly of FIG. 2.

17

18      FIG . 3: is a side view of a second preferred rotor assembly  
19       according to the invention.

20

21      FIG . 3a is an end view of the rotor assembly of FIG. 3.

22

23      FIG. 4: is a plan view of a salient-type stator winding useful in  
24       accordance with the invention.

- 1 FIG. 5: is a plan view of a bifilar-type stator winding useful in  
2 accordance with the invention.  
3
- 4 FIG. 6: is a top view of a stator with a stator mounted hall  
5 device according to the invention.  
6
- 7 FIG. 7: is a side view of the stator with a stator mounted hall  
8 device shown in FIG. 6  
9
- 10 FIGS. 8a-8d: are successive views of a preferred hall effect  
11 device assembly according to the invention.  
12
- 13 FIGS. 9: is a block diagram of the control board electronics  
14 according to the invention.  
15
- 16 FIG. 10: is a detailed schematic of a preferred control board  
17 according to the invention.  
18
- 19 FIG 11: is a schematic of a preferred house keeping power supply  
20 according to the invention.  
21
- 22 FIG 12: is a schematic of a preferred low side drive circuit  
23 including power zener according to the invention.  
24
- 25 FIG: 13: shows an end view of a preferred heat sink assembly  
26 according to the present invention.

1  
2 FIG: 14: shows an side view of the preferred heat sink assembly  
3 shown in FIG. 13.  
4  
5 FIG: 15: shows a top view of a preferred MOSFET retainer bar  
6 according to the present invention.  
7  
8 FIG: 16: shows an side view of the preferred MOSFET retainer bar  
9 shown in FIG. 15.  
10  
11 FIG: 17: Is an upper level flow chart showing the preferred logic  
12 flow of microprocessor software according to the  
13 invention.  
14  
15 FIGS 18-36: Are flow charts showing the preferred logic flow of  
16 the routines and subroutines of the microprocessor  
17 software according to the invention.  
18  
19 **Detailed Description of the Invention:**  
20       With reference to FIG 1, a preferred embodiment of a motor  
21 assembly 1 according to the present invention will be described  
22 in general terms with a more detailed description to follow.  
23       As can be seen, a brushless dc motor 2 is provided having a  
24 conduit box 3, and a heat sink 4 attached thereto. The heat sink  
25 4 is attached to a control board 5 which is fastened inside the  
26 conduit box. The control board includes electronic circuitry for

1 providing rectification of an AC input (not shown) provided through  
2 a cut-away lead exit 6, and for controlling excitation of the  
3 stator field windings. Based on feed back control signals and  
4 user-defined parameters, control electronics on the control board  
5 create and maintain specific motor operating characteristics, e.g.  
6 speed, torque, or current, according to desired specifications. To  
7 allow for dissipation of heat from power elements on the control  
8 board 5, i.e. MOSFETS, a space 7 is provided between the heat sink  
9 4 and the control board 5.

10       Turning to FIGS. 2-2a, the rotor assembly 8 of the motor 2  
11 includes a shaft 9 and a known permanent magnet 10 fixed about a  
12 rotor core 11. Known drive bearings 12, 13, e.g. ball or sleeve  
13 bearings, are provided on either end of the rotor shaft to provide  
14 bearing surfaces. As is known to those skilled in the art, the  
15 permanent magnet 10 may be of either the plastic or ceramic type  
16 depending on desired motor characteristics. Generally, however,  
17 plastic magnets display lower field strength than ceramic magnets.  
18 In the case of a plastic permanent magnet, as shown in FIGS. 2-2a,  
19 the magnet is arranged such that the rotor core 9 outer diameter  
20 matches the magnet 10 inner diameter, and the rotor core and magnet  
21 are flush on both ends 14, 15 of the rotor core 10.

22       Where a ceramic permanent magnet is used, as shown in FIGS. 3-  
23 3a, four sections 16-18 of ceramic magnet are fixed  
24 circumferentially about the rotor core 11 to be 90 degrees offset  
25 from each other. In the preferred embodiment of a 3.3" motor

1       design, a space 20 of about .1" is allowed between each of the  
2       ceramic magnets.

3       As shown in FIGS. 4-6, the stator lamination is preferably a  
4       3.3" shaded-pole configuration 23 produced by FASCO Industries of  
5       Ozark, Missouri. The number of stacks and phase windings can be  
6       varied for individual user application. In the preferred 2-phase  
7       motor design, the phase A 21 and phase B 22 coils may be wound in  
8       a salient-type phase set up, as shown in FIG. 4, or in a bifiler  
9       arrangement whereby the phase A 21 and phase B 22 coils are wound  
10      on each pole, as shown in FIG. 5 (shading bands not shown). The  
11      introduction of bifiler windings on the preferred stator  
12      lamination, as shown, results in a more efficient motor which  
13      converts current to force every 1/4 turn (90 degrees) of the motor.

14      In order to provide reliable control of the operating  
15      characteristics of an electric motor, e.g. speed and torque,  
16      various hall effect devices have been widely used in the art.  
17      Hall effect devices are commonly mounted to an electric motor to  
18      sense the rotational speed of the rotor shaft, and to provide a  
19      control signal representative of the rotational speed for  
20      controlling the operating characteristics of the motor. In the  
21      present invention, the output of a hall device, which is  
22      representative of rotational speed, is supplied to the control  
23      board as a feedback control signal.

24      A crucial factor for efficient motor control, however, is  
25      stable orientation of a hall effect device to the stator windings  
26      of the motor. With the stability being a major concern, keeping in

1 mind production and efficiency, a stator mounted hall assembly 24,  
2 as shown in FIGS 6 and 7 is preferred. Correct hall device 25  
3 orientation requires proper positioning of the device within the  
4 assembly housing 24, as well as stable placement of the assembly 24  
5 on the stator 23.

6 As shown in Fig 6, it has been determined that a fifteen  
7 degree hall device lead angle A relative to the stator windings  
8 (not shown) produces the most efficient motor operation. The  
9 assembly 24 provides for this lead angle with the hall device 25  
10 placed off-center with respect to successive stator poles 26-29.  
11 This arrangement combined with direct stator-mounting of the  
12 assembly 24, virtually guarantees stable orientation of the hall  
13 device to the windings. Advantageously, the assembly is unaffected  
14 by alignment of sleeve, endplate or other peripheral motor  
15 components.

16 Referring also to FIG 6, the assembly includes male-end  
17 connector legs 30,31 with respect to the stator, conveniently  
18 taking advantage of existing female-end conversions, specifically  
19 a stator ground hole 32 and the space 33 between stator lamination.  
20 A preferred hall device assembly is shown in FIGS. 8A-8D. As can  
21 be seen in FIG. 8A, the assembly includes a first leg 30 which is  
22 adapted to securely fit into the stator ground hole 32 (FIG. 6),  
23 and a second T-shaped leg 31 adapted to fit in the space 33 (FIG.  
24 6) between the stator lamination. The vertical portion 34 of the  
25 T-shaped leg 31 fits in the space 33 between the lamination, while

1       the horizontal 35 portion rests against the inner surfaces 37 and  
2       38 of the stator lamination.

3       The male-female attachment of the hall device assembly 24 to  
4       the stator 23 eliminates the need to affix the assembly with  
5       rivets, screws, or other secondary devices. Attachment and removal  
6       of the hall assembly are swift and convenient. In fact, the  
7       assembly can be mounted using either ground hole 32 or 39 (FIG. 6)  
8       or placed on the opposite side of the stator 40 for reversible  
9       motor application. Finally, the hall device 25 is contained within  
10      the confines of the plastic assembly itself eliminating the  
11      possibility of direct damage to the pliable leads 41.

12      Turning again to FIG. 1, the conduit box 3 may be formed in  
13      two embodiments depending on the space requirements of the  
14      application. The first embodiment is a metal or plastic conduit  
15      box 3 fastened to the motor sleeve, as shown in FIG. 1. The  
16      arcuate bottom 42 of the conduit box rests against the motor sleeve  
17      43 and is fastened thereto. A cut-away lead exit 6 is also  
18      provided for connecting AC input leads to the control board 5  
19      through an appropriate connector. In the case of a metal conduit  
20      box, the box is spot welded to the sleeve 2, and the heat sink 4 is  
21      attached to the box with screws through screw holes 44 on either  
22      side 45, 46 of the box. An ABS plastic conduit box may be attached  
23      to the motor sleeve via two weld bolts with the heat sink snapped  
24      in place in the top of the box. The second conduit box embodiment  
25      is a stand-alone conduit box (not shown), and is used when remote  
26      electronics are necessary. The stand-alone conduit box is

1 connected to the motor by 18-24" leads, and can be made from either  
2 plastic or metal.

3 In either embodiment of the conduit box, the control board 5  
4 is contained within the conduit box 3, and is attached to the heat  
5 sink 4. A block diagram of the preferred control board design is  
6 provided in FIG. 9. The operation of the control board electronics  
7 will be described, first in general terms, with a more detailed  
8 description to follow.

9 As shown in FIG. 9, the AC input 47 passes to the AC-DC  
10 converter 48 which is preferably a known bridge rectifier. Ripple  
11 in the DC output 49 of the AC-DC converter is filtered by a  
12 filtering network 50 comprising a simple parallel connected  
13 capacitor. The filtered DC signal 51 is provided to the motor as  
14 a common connection, and as an input to the housekeeping power  
15 supply 52. The housekeeping power supply, as will be described in  
16 detail below, creates and maintains a steady +5V DC signal 53 for  
17 operation of the controller electronics 54.

18 The controller electronics 54 include a microprocessor which  
19 is programmed to create and maintain desired operating  
20 characteristics and motor control from user defined input 65. The  
21 controller electronics 54 provide input 62, 63 to the low side  
22 drive electronics 55 to control the phase A and phase B winding  
23 currents 56, 57 provided by the lowside drive electronics to the  
24 respective stator windings 58, 59. The rotational speed of the  
25 rotor shaft is monitored via a hall device 60 which provides the  
26 controller electronics 54 with a feedback signal 61 representative

1 the rotational speed. Also, a feedback signal 64 is taken from the  
2 lowside drive electronics 55 which is representative of the current  
3 output to the motor. A microprocessor in the controller  
4 electronics controls the phase A and phase B winding currents using  
5 the combination of these two feedback signals 61, 64 and pre-  
6 programmed, user-identified operating characteristics.

7 Turning now to FIG. 10, there is shown a detailed schematic of  
8 a preferred embodiment of a control board according to the  
9 invention. As shown, the AC input 47 is received into the control  
10 board and rectified by a known AC-DC converter 48, preferably a  
11 bridge rectifier. Ripple in the rectified AC is filtered by a  
12 known filter network, e.g. by means of capacitor. The DC output of  
13 the rectifier at 51 is provided as a common connection 66 to the  
14 motor, and as the input to a house-keeping power supply (HKPS)  
15 circuit 52.

16 Including the HKPS 52 in the design of the control board  
17 eliminates the need for a step-down transformer as required in  
18 prior art designs. As a result, the motor can be directly  
19 connected to a wall outlet without the use of a costly, space-  
20 consuming AC power supply. In addition, the HKPS can handle a  
21 large variation in line (input) voltage without degrading the logic  
22 (output) voltage. This feat can be done without the addition of a  
23 secondary regulator.

24 Referring particularly to FIG. 11, the input voltage at 51 to  
25 the HKPS 52 is considered 200 VDC maximum (160VDC with load). The  
26 resistor R2, preferably a 150k ohm/.25W device, limits the current

1 through the zenor diode VR1 which is a rated 500 mW element. The  
2 zenor diode VR1 must conduct at a voltage high enough to overcome  
3 the threshold voltage of the gate 67 of the MOSFET Q1, which for  
4 the preferred IRF630 MOSFET is 2-4 VDC. The zenor diode is  
5 reverse-biased at 9.1VDC. With the zenor diode in conduction, the  
6 MOSFET Q1 is on, and the drain 68 current flows allowing the  
7 capacitor C2 to charge.

8 The capacitor C2 will charge until the MOSFET source 69  
9 voltage is greater than the gate voltage (non-conduction MOSFET  
10 mode). This occurs when the combination of the gate threshold  
11 voltage of the MOSFET and the potential voltage of the capacitor C2  
12 is greater than the zenor diode VR1 voltage. With the MOSFET off,  
13 the capacitor will discharge a discrete amount, until the MOSFET  
14 source voltage is less than the gate voltage. At this point, the  
15 MOSFET is on, and the process repeats itself. The diode CR2 at the  
16 drain 68 prevents back current from discharging the capacitor C2 in  
17 the event that the input voltage is lost. The end result is a  
18 MOSFET Q1 that continually switches on and off maintaining a  
19 relatively stable 5 VDC output 70 for the controller electronics.

20 The 5 VDC output is achieved as long as the input signal falls  
21 within an acceptable range determined by the performance parameters  
22 of the circuit elements, specifically the MOSFET. In this  
23 configuration, for an IRF630 MOSFET the input voltage can vary from  
24 17-200VDC. As a result, the HKPS can handle a large variation in  
25 line voltage without effecting logic voltage. Another advantage of

1 this HKPS is its ability to "step-down" such a wide range of  
2 voltage (i.e. 200VDC to 5VDC).

3 Referring again to FIG. 10, several microprocessors can be  
4 used according to the invention. Preferably, however a PIC16C71  
5 microprocessor 12 available from Microchip Corporation of Chandler,  
6 Arizona, USA is used based on cost, efficiency, and performance  
7 characteristics. The microprocessor input 65 is user defined  
8 depending on the application. The user i/o interface 52 is active  
9 low with select bits provided to the user for control of motor  
10 characteristics, e.g. speed and torque. Based on the user i/o, the  
11 microprocessor 12 controls the phase A 71 and phase B 72 outputs to  
12 achieve the desired operating characteristics. The software for  
13 the microprocessor logic used to control the phase A and phase B  
14 outputs will be discussed in detail below.

15 Referring particularly to FIG. 12, the phase A 70 and phase B  
16 71 outputs of the microprocessor 12 are provided as inputs to the  
17 gates 73, 74 of driving MOSFETS Q4, Q3 in the low side drive  
18 electronics 55. Since there are two phases to the motor, two  
19 driving MOSFETs Q4, Q3 are used, one for each phase. Two diodes  
20 CR3, CR4 are placed in series with each driving MOSFET to prevent  
21 back current from one driving MOSFET to the other.

22 A power zenor diode 75 is incorporated into the low side drive  
23 electronics 55 to account for a momentary increase of voltage (1200  
24 VDC) at the drains 76, 77 of the driving MOSFETS due to back EMF  
25 caused by the switching action of the driving MOSFETS. This effect  
26 is inherent in a bifilar wound motor (phase-shared magnetic fields

1 due to opposed windings on each pole) unlike a motor with salient  
2 poles. In order for the driving MOSFETS to control drain current,  
3 the back EMF must be managed. This management concept is  
4 accomplished with the application of a zenor diode configuration  
5 75 capable of high-power dissipation (hence the name "power  
6 zenor").

7 The voltage at the source of the MOSFETS is 200VDC maximum  
8 (160VDC nominal); approximately 400VDC is the maximum desired  
9 output voltage at the drain. To accomplish this, the zenor diode  
10 VR2 is rated at 180V. While the driving MOSFETs Q3, Q4 are  
11 operating (one is off and the other is on) there is no current  
12 through the power zenor 75 (i.e. voltage at power zenor MOSFET Q2  
13 drain 78 is less than 400VDC).

14 At the point when the driving MOSFETS switch from on to off,  
15 the voltage at the drain 78 of the power zenor MOSFET Q2 rapidly  
16 increases resulting in a reverse-biased zenor diode VR2. Since the  
17 zenor diode voltage VR2 is constant at this time, any excessive  
18 voltage applied to the MOSFET drain is dissipated through the  
19 MOSFET rather than the zenor diode VR2. The zenor diode VR2 clamps  
20 the voltage at the drain 78 equal to the zenor diode voltage plus  
21 the 200VDC at the source (gate threshold voltage must be considered  
22 but at 2-4VDC is negligible). The power zenor 75 conducts until  
23 the switching action of the driving MOSFET is accomplished and the  
24 resulting increased voltage resonates back to an acceptable level.

25 Where a 115 VAC input (i.e. from a wall outlet) is not  
26 desired, a power zenor configuration 75 may not be needed. For

1       input voltages up to 100VDC or 70VAC a different configuration can  
2       be utilized. Because the zenor diode VR2 can handle power  
3       dissipation at this level, the extra MOSFET Q2 in the power zenor  
4       configuration is no longer needed. Instead, a zenor diode VR2  
5       (with accompanying protection diode) is placed between the drain  
6       and gate of each driving MOSFET Q3,Q4.

7           In the preferred embodiment of the invention, the aluminum  
8       heat sink 4 is provided for dissipating heat produced by the power  
9       devices (i.e. four MOSFETs Q1-Q4) contained on the motor control  
10      board 5. Referring to FIGS 13-14, it has been found that  
11      production efficiency and overall cost is improved by aligning all  
12      four MOSFETS Q1-Q4 above and to one side of the control board 5.  
13      The heat sink 4 dissipates most of the heat through the fins 79  
14      below the board 5 which is positioned within a slots 80, 81 on  
15      either side of the heat sink. One side 82 of the heat sink extends  
16      above one side of the control board 5 for MOSFET attachment.

17           With this configuration, several desirable results are  
18       accomplished. First, with the MOSFETs above the control board, the  
19       board itself can be wave-soldered with the MOSFETs in place. This  
20       eliminates the additional step of hand-soldering the MOSFETs to the  
21       board (as is necessary when these devices are attached to the heat  
22       sink below the board). Second, the MOSFETs Q1-Q4 are connected to  
23       the heat sink via a Teflon plastic or aluminum retainer bar 83,  
24       shown particularly in FIGS. 15 and 16. The MOSFETs are secured to  
25       the heat sink underneath extension 84 of the bar 83. The bar, in  
26       turn, is held in place against the heatsink 4 with only two screws

1 through holes 85 , 86 in the bar and holes 87,88 in the heat sink  
2 4, instead of an individual screw for each MOSFET. Finally, by  
3 placing the MOSFETs in this above-board configuration additional  
4 space is afforded to the control board for circuit design.

5 Turning again to FIG. 10, to control the phase A and phase B  
6 winding currents, the microprocessor 12 receives feedback from the  
7 hall device 60, and from a lowpass filter/op amp circuit 89. The  
8 hall device 60 provides a signal which is representative of the  
9 rotational speed of the rotor shaft 9. The low pass filter is  
10 connected to the sources 90, 91 of the driving MOSFETS Q4, Q3 to  
11 obtain a current signal at 92 representative of the driving  
12 current. The difference between this signal, and a signal at 95  
13 representative of the operational amplifier output at 93, is  
14 amplified by the op amp 94 and provided as a feedback input to the  
15 microprocessor 12. The output of the op amp at 93 increases,  
16 therefore, with increasing drive current. The microprocessor 12  
17 uses the hall device signal 61 and the op amp feed back signal at  
18 93 to control the torque (rotational speed x current), rotational  
19 speed, and/or current of the motor by adjusting the phase A and  
20 phase A outputs in response to sensed changes.

21 FIGS. 17 is a basic flow chart for the software which is  
22 preferably used in the microprocessor 12 to control motor  
23 parameters. The software may be customized to control several  
24 motor parameters according to user desired performance.

25 With reference to FIG. 17, upon power up 90 the microprocessor  
26 self test 91 is initiated by the software to verify the integrity

1       of the microprocessor.     The software then performs an  
2 initialization routine 92, a three pole switch routine 93 to  
3 determine the status of a user defined/controlled switch inputs  
4 223, and a soft start routine 94 to slowly ramp up the speed of the  
5 rotor. A Main Setup routine 95 is then initiated to set flags and  
6 registers used in the main routine 96. The main routine 96  
7 executes various user specific applications 97 in combination with  
8 standard subroutines 98 to control speed, torque, current, and/or  
9 volume flow output 225 based on user input and feedback signals  
10 224.

11       The timer interrupt routine 99 is used in connection with the  
12 main routine and the soft start routine to provide a timed  
13 interrupt for incrementing counters, saving data, setting flags,  
14 etc. The timer interrupt routine includes a timer subroutine 100  
15 for determining necessary speed adjustments and performing  
16 necessary speed changes. A Pulse Width Modulation (PWM) Interrupt  
17 101 operates in connection with the Main Routine for counting hall  
18 edges and controlling the pulsing of the phases for speed control.  
19 Finally, motor shut down is performed by either a hard brake 102  
20 routine or an instant shut down routine 103. The Hard Brake 102  
21 routine responds to the hard brake flag to shut down the motor with  
22 very little coasting. The instant shutdown 103 routine allows the  
23 motor to coast to a stop.

24       Two assembly language code program listings specific to the  
25 preferred Microchip PIC16C71 are attached hereto. These programs  
26 listings contain preferred versions of the routines and

1 applications described below. One program is designed to implement  
2 a constant CFM in a particular application, and the other is  
3 designed to create a constant RPM. Changes in these specific  
4 applications are required based on required operating  
5 characteristics, as will be apparent to those skilled in the art.  
6 Nonetheless, the standard routines set forth in the listings remain  
7 relatively unchanged from application to application.

8 Turning now to FIGS. 18-36, a detailed explanation of each  
9 routine will follow. With reference now to FIG. 18, the Initialize  
10 Routine 92 defines constants 104, defines register addresses 105,  
11 defines interrupts and code origins 106, defines I/O ports 107,  
12 defines interrupts and the pre-scaler 108, initializes the timer  
13 109, and clears all remaining registers 110.

14 The switch routine 93, as shown in FIG. 19, is initiated to  
15 determine the status of the user input for controlling motor start  
16 up. The SELECT, SELECT INPUT 0, and SELECT INPUT 1 inputs to the  
17 Switch Routine are user defined based on the application.  
18 Alternatively, an external three-position switch may be provided to  
19 allow external control of these processor inputs for user control  
20 of motor operation.

21 The switch routine first determines if the select input 115  
22 has been set low. If not, then soft start routine 94 is initiated.  
23 If so, then if speed input 0 is low and speed input 1 is high 116,  
24 then the soft start routine 94 is initiated. Otherwise, the  
25 routine ends 114 and the motor will not start. For virtually all  
26 applications, the motor/controller will not operate when the user

1 controlled three-position switch is in the middle position, and  
2 will operate in the low or high positions.

3       The soft start routine 94 follows the switch routine 94 and  
4 slowly ramps up the motor from a stopped rotor position by pulsing  
5 the phases on and off (PWM), resulting in a softer start up and  
6 less noise. An added feature is an increasing variable torque  
7 factor which allows the motor to start up under heavier loads and  
8 colder temperatures. Speed is increased in this routine by  
9 changing the duty cycle of the PWM, thus creating more torque as  
10 the speed of the motor increases.

11       Referring to FIGS. 20 and 21, the soft start routine begins by  
12 initializing the PWM counter 117. The hall signal 119 is then  
13 checked 118 for a high or low condition. If the hall signal is  
14 low, phase A processor output is selected for pulse width  
15 modulation (PWM) 120, phase B output 121 is set low, and rising  
16 hall edge detection is set 123. If the hall signal is high,  
17 phase B is selected for PWM 124, phase A is set low 125, and  
18 falling hall edge detection is set 126.

19       One PWM is then performed 127. The bus voltage input 129 is  
20 then checked. If the bus voltage is below minimum 128, then a  
21 timeout flag is set 130. If it is not below minimum, then the  
22 current sense input 131 is checked 132 to determine if it is above  
23 maximum. If the maximum current sense is exceeded a timeout flag  
24 is set 133.

25       Next, the hall signal 134 is checked 135 to determine if the  
26 next hall edge has been received. If so, the routine loops back to

1       the beginning. If not, the PWM counter is checked 136 to see if it  
2       is equal to zero. If PWM counter does not equal zero, the routine  
3       loops back to perform one PWM. If the PWM counter equals zero, the  
4       soft start timer is checked 137 to determine if it has expired. If  
5       the timer has expired, the routine proceeds to the main setup 95.  
6       If not, the routine loops back to the beginning.

7           A Timer Interrupt routine 99 is used in connection with the  
8       main routine and the soft start routine 94. The Interrupt is  
9       activated based on the 1:256 prescaler, which means it occurs every  
10      65,536 microseconds. Every fifteenth time through this interrupt,  
11      approximately one second has elapsed, thus triggering the tasks of  
12      incrementing the seconds counter, saving the hall edges per second,  
13      and setting a calculation flag (if necessary), and calling a timer  
14      subroutine (if necessary). The timer subroutine performs a  
15      comparison of registers to determine rotor speed adjustment. This  
16      routine is called from the timer interrupt and performs the  
17      necessary speed changes.

18           Referring to FIG. 22, the Timer Interrupt routine flow depends  
19       on the status 138 of the soft start routine 94. While the soft  
20       start is proceeding, the timer interrupt routine checks to  
21       determine whether the motor is in the desired operational window  
22       139 . If not, the routine increments or decrements the PWM counter  
23       appropriately 140 and sets the timeout flag 141. If the  
24       operational window has been, then the timeout routine determines if  
25       one second has expired 144. If not, the routine ends 146. If one

1 second has not expired, the speed and set up variables are saved,  
2 and the timer is reset before the routine ends 145.

3 If the soft start routine is finished, then the Timer  
4 Interrupt routine checks the PWM counter to determine if it is less  
5 than or equal to the destination PWM value 142. If it is not, then  
6 the PWM counter is decremented 143. If it is, then the Timeout  
7 Routine determines if one second has expired 144. If not, the  
8 routine ends 146. If one second has not expired, the speed and set  
9 up variables are saved, and the timer is reset 145 before the  
10 routine ends 146. A timer subroutine is called from the timer  
11 interrupt in most applications to perform the necessary speed  
12 changes. Referring to FIG. 23, the timer subroutine compares a  
13 predefined theoretical value of the current, bus voltage, torque,  
14 speed, or any calculated value, to the actual value calculated  
15 value to determine the speed change 147. If the values are equal  
16 148, then the routine returns to the Timer Interrupt routine 99.  
17 If the theoretical and actual values are different, then the Timer  
18 Subroutine determines the increment or decrement to speed via PWM  
19 149. The necessary changes to the PWM are performed 150, and the  
20 flow returns to the Timer Interrupt routine 99.

21 Referring to FIG. 24, a Main Setup routine 95 follows the soft  
22 start 94 to prepare for the main processing depending on user  
23 defined operating characteristics. In this routine, user defined  
24 variables are initialized 151 the flow proceeds to the main  
25 processing routine 96. Also, flags are set, and the global

1 interrupt enable is set. Flags may be set here for OWC output  
2 on/off and for a hard brake or natural coast stop.

3 Referring again to FIG. 17, the Main Routine contains 96  
4 several routines 98 and calls various applications 97 according to  
5 desired user specifications. Depending on user requirements,  
6 within the main routine may be applications for creating variable  
7 speed 152, selectable speeds 161, variable current 153, variable  
8 torque 154, constant speed 160, constant current 156 , constant  
9 torque 155, constant volume flow 158, selectable speeds 161, and/or  
10 for following a curve or table may be accessed 159. Routines used  
11 within the main routine in connection with these applications  
12 include, a Pulse Width Modulation (PWM) Interrupt Routine 101, a  
13 Maximum/Minimum Routine 162, Transition Routine 163, A/D Routine  
14 164, Calculation Routine 165, and a Table Lookup Routine 166.

15 The PWM Interrupt routine 101 activates a motor controlling  
16 Interrupt on a hall edge. The first task is to determine the  
17 correct phase to fire. The selected phase is then pulsed on and  
18 off eight times, with forty-microsecond pulses, to soften current  
19 change. For speed control, the phase is off for a varying amount  
20 of time, and then turned on off the remaining time until another  
21 hall edge is received. A count of hall edges per second is also  
22 done here.

23 Referring to FIGS. 25 and 26, the PWM interrupt 101 begins  
24 with initialization of the PWM counter 167. The hall signal 169 is  
25 then checked for a high or low condition 168. If the hall signal  
26 is low, phase A processor output is selected for pulse width

1 modulation (PWM) 173, phase B output is set low 174, and rising  
2 hall edge detection is set 174. If the hall signal is high, phase  
3 B is selected for PWM 170, phase A is set low 171, and falling hall  
4 edge detection is set 172. In some applications where full power  
5 is not desired, an additional step 178 is performed to determine if  
6 the user has required a skip of phase PWM for a preset number of  
7 hall edges. If so, the flow proceeds to the end of the routine  
8 190.

9 One PWM is then performed 180. The bus voltage input is then  
10 checked 182. If the bus voltage 181 is below minimum, then a  
11 timeout flag is set 183. If it is not below minimum, then the  
12 current sense input 184 is checked 185 to determine if it is above  
13 maximum. If the maximum current sense is exceeded a timeout flag  
14 is set 186.

15 Next, the hall signal 188 is checked 187 to determine if the  
16 next hall edge has been received. If so, the routine loops back to  
17 the beginning 101. If not, the PWM counter is checked to see if it  
18 is equal to zero 189. If PWM counter does not equal zero, the  
19 routine loops back to perform one PWM 180. If the PWM counter  
20 equals zero, the PWM Interrupt Routine ends 190.

21 The Minimum/Maximum routine 162 is performed to check to see  
22 that the bus voltage, current sense, and rotor speed are within a  
23 particular range, otherwise an instant shutdown is performed.  
24 These checkpoints are determined by the user. Referring to FIG.  
25 27, the Minimum/Maximum routine 162 checks the minimum and maximum  
26 limits on bus voltage, speed and current 191. If a limit has been

1        exceeded 192 then the Instant Shutdown routine is performed 103.  
2        If the limit has not been exceeded, then flow returns to the Main  
3        Routine 96.

4            The instant shutdown routine 103 turns the phase outputs off  
5        and waits for the switch to be cycled off then back on, which  
6        restarts the microprocessor. The motor will coast to a stop unless  
7        the switch is cycled. Referring to FIG. 28, the instant shutdown  
8        routine 103 disables all interrupts 193, turns both phases 196 off  
9        194, and sets OWC 197 low 195. Instant Shutdown then determines  
10      198 if Speed Input 0 is low and Speed Input 1 199 is low. If so,  
11      then the flow returns to beginning 90 to restart the microprocesor.  
12      If not, then flow loops back to the beginning of the Instant  
13      Shutdown routine 103.

14           A Hard Brake Routine may also be called by the main routine in  
15       some application to turn off both phases if a hard brake flag is  
16       set. Opposite phases are selected and pulsed on and off every  
17       three microseconds to slow the rotor down dramatically. The phases  
18       are alternated between hall edges until the time elapsed between  
19       two hall edges is large enough to represent a stopped rotor. Very  
20       little coasting occurs, if any.

21           Referring to FIG. 29, the Hard Brake Routine 102 turns  
22       interrupts off 200, and checks the hall position 201. Opposite  
23       phases are pulsed in three microsecond pulses 202. When the time  
24       between two hall edges is between a user-defined, pre-set number of  
25       counts, the shut down routine is executed 203.

1       The transition routine controls flags and handles delays  
2 between switching. When switching to the off position after soft  
3 start, there is a two second delay before instant shutdown. When  
4 switching from high to low or vice versa, there is an eight second  
5 transition period where calculations and some tests are skipped to  
6 allow motor performance changes. In some applications, set points  
7 are needed, such as constant speed, constant current, and constant  
8 torque. These are set when an optional three-position user  
9 operated switch is thrown.

10      Referring to FIG. 30, the Transition Routine 163 first  
11 determines whether the three-position switch 205 has changed  
12 position 210. If not, then the transition flag is cleared after 8  
13 seconds 209 and flow is continued to the main routine 96. If the  
14 switch position has changed, the Transition routine sets  
15 variables/flags to reflect the change 204, sets the off flag if the  
16 switch is in the off position 206, clears the off flag if the  
17 switch is changed from "off" in 2 seconds or less 207, and  
18 transfers flow to the shutdown routine if the switch is off for  
19 more than 2 seconds 208.

20      The A/D routine performs and controls A/D conversion of analog  
21 input signals. All A/D conversions are based on the bits in the  
22 PCYCLE register which starts at zero. If the LSB bit is zero, the  
23 last A/D conversion is saved and the next conversion is setup based  
24 on bits one and two. If the LSB bit is one, the twenty microsecond  
25 A/D conversion is started. In all cases the PCYCLE register is

1 incremented for proper execution on the following loop through the  
2 main routine.

3 Referring to FIG. 34, the A/D routine 164 checks the PCYCLE  
4 value and branches 212 based on PCYCLE to: (1) save speed/setup  
5 bus voltage; (2) save speed/setup current; (3) save bus  
6 voltage/setup speed; (4) save current/setup speed; or (5) perform  
7 conversion. The A/D routine then increments PCYCLE 213 and returns  
8 flow to the Main Routine 96.

9 The calculation routine is required in applications such as  
10 the constant and variable torque applications. The calculation  
11 routine can be located anywhere in the main routine but is usually  
12 found after the A/D routine. A calculation flag is set in the  
13 Timer Interrupt to determine how often the calculation is to be  
14 performed. Calculations are usually not performed during a switch  
15 transition period. Referring to FIG. 35, the flow of the  
16 calculation routine 165 for torque is shown. First, the  
17 Calculation Routine loads the number of Hall edges per second and  
18 stores the value 214. The routine then loads the number of current  
19 base on an A/D conversion 215, and multiplies the hall edges by the  
20 current to determine torque 216. Flow is then passe back to the  
21 main routine 96.

22 The table look up routine is required by some applications to  
23 determine the proper rotor speed, such as applications for creating  
24 constant volume flow, following a curve/table, or possibly  
25 selectable speeds. Referring to FIG. 36, the Table Lookup routine  
26 loads or calculates an index 217 and then calls a table 218. The

1       table location is loaded 219, the index value is added 220 and  
2       moved into the program counter 221, and flow is returned to the  
3       table lookup 22. The table lookup then returns flow to the Main  
4       Routine 96.

5       Turning again to FIG 17, a brief description of the user  
6       defined applications 97 executed in the main routine with the above  
7       described routines is provided below

8       Variable speed - Speed is adjusted through a potentiometer  
9       giving a true variable speed within the resolution of the  
10      potentiometer A/D conversion. The potentiometer value is saved  
11      directly into the varying delay register so that no timer  
12      subroutine is needed to determine speed change. Calculations or  
13      calculation flags are not needed.

14      Variable current - Current is adjusted through a potentiometer  
15      giving a variable current within the resolution of the  
16      potentiometer and current A/D conversions. The potentiometer value  
17      is saved directly into the theoretical current register and is  
18      compared to the actual current to determine speed change. This  
19      requires a timer subroutine. Calculations or calculation flags are  
20      not needed.

21      Variable Torque - Torque is adjusted through a potentiometer  
22      giving a variable torque within the resolution of the potentiometer  
23      current and current A/D conversions and of the error counting the  
24      hall edges per second. The potentiometer value is saved directly  
25      into the theoretical torque register and is compared to the  
26      calculated torque value to determine speed change. This requires

1       a timer subroutine, a torque calculation routine, and a calculation  
2       flag.

3           Constant Speed - Speed is set in the transition routine where  
4       the value is saved directly into the theoretical speed register and  
5       is compared to the actual speed value to determine speed change.  
6       This requires a timer subroutine. Calculations, calculation flags,  
7       and potentiometer A/D conversion are not needed.

8           Constant Current - Current is set in the transition routine  
9       where the value is saved directly into the theoretical current  
10      register and is compared to the actual current value to determine  
11      current change. This requires a timer subroutine. Calculations or  
12      calculation flags are not needed.

13          Constant Torque - Torque is set in the transition routine  
14       where the value is saved directly into the theoretical torque  
15      register and is compared to the calculated torque value to  
16      determine speed change. This requires a timer subroutine, a torque  
17      calculation routine, and a calculation flag.

18          Constant Volume Flow - For constant CFM, actual torque is  
19       calculated in the main routine. This value is used in a table  
20      lookup to determine the theoretical rotor speed for that particular  
21      torque. This theoretical speed is compared to the actual speed  
22      value to determine speed change. This requires a timer subroutine,  
23      a torque calculation subroutine, a calculation flag, and a torque  
24      /speed lookup table.

25          Selectable Speeds - By using a lookup table and several input  
26      lines, a variety of combinations can be used for selectable speed

1 control. The selected speed value is compared to the actual speed  
2 value to determine speed change. This routine requires a timer  
3 subroutine. Calculations, calculation flags, and potentiometer  
4 conversion are not needed. This routine is very similar to  
5 constant speed.

6 Follow Curve/Table - Any equation can be put in table format  
7 and followed based upon speed, current, torque, etc. One dependent  
8 value is found in a table based on another independent value.  
9 These values can be compared to determine speed change. This  
10 requires a timer subroutine, possibly a calculation routine, a  
11 calculation flag, and a lookup table.

12 The embodiments which have been described herein are but some  
13 of the several which utilize this invention and are set forth here  
14 by way of illustration but not of limitation. It is obvious that  
15 many other embodiments which will be readily apparent to those  
16 skilled in the art may be made without departing materially from  
17 the spirit and scope of this invention.

18

19

```

; DESCRIPTION Two Speed Constant RPM -
; - Soft start with progressive torque
; - Hard brake or natural coast stop
; - Max/Min speed, bus voltage, and current sense monitor
; - Two Speed Settings (Low/High)
; - Low Speed Every Other Phase/High Speed Every Phase
; - Variable Speed Compensation
;

;-----  

; REGISTER/BIT ADDRESSES
;  

;-----  

; Predefined Register Addresses
;  

LIST P=16C71           ;Identifies PIC16C71
;  

PIC71    equ    0h          ;16C71 Special-purpose registers  

INDF    equ    0h  

RTCC    equ    1h  

OPTION  equ    01h  

PCL    equ    2h  

STATUS  equ    1h  

FSR    equ    4h  

PORTA   equ    5h  

PORTB   equ    6h  

PORTC   equ    7h  

ADCON0  equ    8h  

ADCON1  equ    08h  

ADRES   equ    9h  

PCLATH  equ    0Ah  

INTCON  equ    0Bh  

TRISA   equ    05h  

TRISB   equ    06h  

IRP    equ    7h          ;16C71 STATUS bits  

RP1    equ    6h  

RPO    equ    5h  

TO     equ    4h  

PD     equ    3h  

Z      equ    2h  

DC     equ    1h  

CARRY  equ    0h  

GIE    equ    7h          ;16C71 INTCON register bits  

ADIE   equ    6h  

TOIE   equ    5h  

XTE    equ    4h  

RBIE   equ    3h  

TOIF   equ    2h  

INTF   equ    1h  

RBIF   equ    0h  

ADCS1  equ    7h          ;16C71 ADCN0 register bits  

ADCS0  equ    6h  

CHS1   equ    4h  

CHS0   equ    3h  

ADGO   equ    2h  

ADIF   equ    1h  

ADCN   equ    0h  

PCFG1  equ    1h          ;16C71 ADCN1 register bits  

PCFG0  equ    0h  

INTE00  equ    0h          ;16C71 OPTION register bits
;  

; PORTB register bits
;  

HALL   equ    0h          ;Hall Signal Input  

SSIN1  equ    1h          ;Set Speed Input MSB  

SSIN0  equ    0h          ;Set Speed Input LSB  

OWC   equ    2h          ;Operational Window Control  

PHE   equ    4h          ;Phase B Output  

PAE   equ    5h          ;Phase A Output  

SELECT equ    6h          ;Speed Select
;  

;BITREG register bits
;
```

```

;
OFFSWTC equ 0h          ;Off Switch Flag
TRANS8 equ 1h           ;8 Second Transition Flag
HIGHLOW equ 2h           ;High/Low Speed Flag
SLOWFST equ 3h           ;Slow Or Fast Speed Transition Flag
HBRAKE equ 4h            ;Hard Brake On/Off Flag
ONCFLAG equ 6h            ;ONC Flag
SFTSTRT equ 7h           ;Soft Start Flag
;
;BITREG2 register bits
;
;
;Define General Purpose Registers
;
H_BYTE equ 0Ch          ;High Byte Of 16-bit Result
L_BYTE equ 0Dh          ;Low Byte Of 16-bit Result
PCNTRO equ 10h          ;PWM Counter LSB
PSTORO equ 11h          ;PWM Storage LSB
THEO1 equ 12h           ;Theoretical MSB
THEO0 equ 13h           ;Theoretical LSB
PCYCLE equ 14h          ;PWM Cycle Counter
TCNTR equ 15h           ;Timer Counter
SECCNTR equ 16h          ;Seconds Counter
HCNTR1 equ 17h          ;Hall Edges Counter MSB
HCNTR0 equ 18h          ;Hall Edges Counter LSB
HSTOR1 equ 19h          ;Hall Edges Storage MSB
HSTOR0 equ 1Ah           ;Hall Edges Storage LSB
BV equ 1Bh              ;Bus Voltage Value (A/D)
CSACTL equ 1Ch           ;Current Sense Actual Value
BITREG equ 1Dh           ;Bit Register For Loop Control
BITREG2 equ 1Eh          ;Bit Register For Loop Control 2
PHASE equ 1Fh           ;Select Phase A Or B
WSAV equ 20h             ;Save Value Of W Register
STATSAV equ 21h          ;Save Value Of Status Register
SPDADJ equ 22h           ;Speed Adjustment
TEMP1 equ 23h           ;Temporary Register 1
TEMP2 equ 24h           ;Temporary Register 2
CNTR1 equ 25h           ;General Counter 1
CNTR2 equ 26h           ;General Counter 2
;
;
;
-----
;
; CODE ORIGINS
;
-----
;
;Establish Program Origin And Interrupt Vectors
;
    org 00h      ;reset vector
    goto Start
;
    org 04h      ;interrupt vector
    goto Int_V
;
    org 10h      ;start of code
;
;
;
-----
;
; INITIALIZE REGISTERS
;
-----
;
;Initialize Page 1 Registers
;
Start  bsf STATUS,RFC   ;select pg1 registers
        movlw b'00011111' ;set port A I/O (1=input)
        movwf TRISA         ;send values to port A
        movlw b'11000111' ;set port B I/O (1=input)
        movwf TRISE         ;send values to port B
        movlw 1              ;change AIN3 to Vref for A/C
        movwf ADCON1         ;set ADCON1 register
        movlw b'01000111' ;select prescaler, timer and INT edges
        movwf OPTION         ;set OPTION register
;
;Clear Variables

```

```

;
bcf STATUS,RFO      ;select pg0 registers
clrf PORTA           ;clear
clrf PORTB           ;clear
clrf ADCONO          ;clear
clrf ADRES            ;clear
clrf PCLATH           ;clear
clrf INTCON           ;clear
clrf TCNTR            ;clear
clrf PCNTRO           ;clear
clrf PSTORO           ;clear
clrf THEO1            ;clear
clrf THEO0            ;clear
clrf HCNTRO           ;clear
clrf HCNTR1           ;clear
clrf HSTORO           ;clear
clrf HSTOR1           ;clear
clrf CSACTL           ;clear
clrf PCYCLE           ;clear
clrf PHASE             ;clear
clrf WSAV              ;clear
clrf STASAV           ;clear
clrf SECCNTR           ;clear
clrf BITREG            ;clear
clrf BITREG2           ;clear
clrf TEMP1             ;clear
clrf TEMP2             ;clear
clrf CNTR1             ;clear
clrf CNTR2             ;clear
;
;Initialize Variables
;
movlw .15             ;set timer to approximately 1 sec
movwf TCNTR           ;timer counter
movlw .75             ;set starting value
movwf BV               ;save bus voltage
movwf CSACTL           ;save current sense
;
;
-----
;
; START UP ROUTINE
;
-----
;
;Start Up Switch
;
btfsr PORTB,SELECT    ;check if speed select low
goto Start             ;restart micro
btfsr PORTB,SSINGO    ;wait for speed input LSB to go low
goto Go                ;continue
btfsr PORTB,SSINI     ;wait for speed input MSB to go high
goto Start             ;restart micro
;
;Perform Soft Start With Increasing Torque
;
Go    bsf    BITREG,SFTSTRT  ;set soft start flag
      movlw .191           ;number of hall edges
      movwf TEMP1           ;save into general counter
Halcing  movf  TEMP1,0        ;load N
      movwf PSTORO          ;save increasing torque delay counter
      movlw .5               ;shift right loops
      movwf CNTR1           ;save into general counter
Shifting bcf   STATUS,CARRY  ;clear carry bit
      rrf    PSTORO,1        ;divide by 2
      decfsr CNTR1          ;decrement general counter
      goto Shifting         ;continue shifting
      movf  PSTORO,0        ;load N
      btfsr STATUS,I         ;check if zero
      goto Msetup            ;continue with Main Setup
      moviw .1                ;number of hall edges doubles
      movwf CNTR1           ;save into temporary counter
Halcing1 btissr PORTB,HALL  ;alignment of hall signal and phases
      goto Hall1             ;continue with high hall signals
      bcf   PORTB,PHB        ;turn phase B off
      bcf   PHASE,I           ;select phase A
      bcf   PHASE,1           ;unselect phase B
      bcf   STATUS,RFO        ;select pg0 registers

```

```

        bsf    OPTION,INTEEG :set rising hall edge detect
        bcf    STATUS,RPO  ;select pg0 registers
        goto  Quar_0       ;continue
Hihall   bcf    PORTB,PHA  ;turn phase A off
        bcf    PHASE,5    ;unselect phase A
        bsf    PHASE,4    ;select phase B
        bsf    STATUS,RPO  ;select pgl registers
        bcf    OPTION,INTEDG
        bcf    STATUS,RPO  ;set falling hall edge detect
        movwf TEMP1,0      ;select pg0 registers
        movwf PCNTRO       ;number of pwms (y of x*y)
        movlw .25          ;save into general counter
        movwf TEMP2         ;number of pwms (x of x*y)
        btfsc PORTB,SSINO  ;save into general counter
        goto  Hal_1        ;is speed input LSB low
        btfss  PORTB,SSIN1  ;is speed input MSB high
        goto  Sd_inst      ;no...instant shutdown
Hal_1    movwf PHASE,0      ;load PHASE into W
        xorwf PORTB,1      ;set phase high
        movwf PSTORO,0     ;load W
        sublw .6            ;6-PSTORO delay loops
        movwf CNTR1         ;save into general counter
        decfsz CNTR1       ;decrement general counter
        goto  Del_0         ;continue delay
        movwf PHASE,0      ;load PHASE into W
        xorwf PORTB,1      ;set phase low
        btfsc INTCON,INTF  ;check if hall edge received
        goto  Halrcvd      ;continue
        movwf PSTORO,0     ;load W
        movwf CNTR1         ;save intc general counter
        decfsz CNTR1       ;decrement general counter
        goto  Del_1         ;continue delay
Del_1    decfsz TEMP2       ;decrement general counter
        goto  Hal_1         ;continue pulsing
        decfsz PCNTRO       ;decrement general counter
        goto  Quar_1        ;continue loop
        movwf PHASE,0      ;load PHASE into W
        xorwf PORTB,1      ;set phase high
Edgwait  btfsc PORTB,SSINO  ;is speed input LSB low
        goto  Edg_1         ;continue
        btfss  PORTB,SSIN1  ;is speed input MSB high
        goto  Sd_inst      ;no...instant shutdown
Edg_1    btfss  INTCON,INTF  ;check if hall edge received
        goto  Edgwait      ;continue waiting for hall edge
Halrcvd  bcf    INTCON,INTF  ;clear hall edge if present
        decfsz CNTR2       ;decrement general counter
        goto  Halcnq2      ;continue loop
        decfsz TEMP1        ;decrement general counter
        goto  Halcnq        ;continue loop
;
;Setup For PWM, Timer, And Main Program Processing
;
Mssetup  movlw .25          ;PWM starting point LSB
        movwf PSTORO       ;save in PWM storage LSB
        movwf PCNTRO       ;save into PWM counter LSB
        movlw .15          ;set timer to approximately 1 sec
        movwf TCNTR        ;timer counter
        clrf  SECCNTR       ;clear seconds counter
        movlw b'00110000'   ;clear flags and enable TMRO and PWM
        movwf INTCON        ;write INTCON register
        bsf   INTCON,GIE    ;global enable
        bcf   BITREG,SFTSTRT ;clear soft start flag
        bsf   BITREG,OWCFLAG ;set OWC output
        bcf   BITREG,HBRAKE  ;SET HARD BRAKE
;
;
;
-----
;
; MAIN PROGRAM
;
;
Main
;
;Call A/D Routine
;
        call   Ad_main      ;A/D conversion routine
;

```

```

;Check Bus Voltage & Current Sense Minimum/Maximum
;
    movlw .230      ;current sense absolute maximum @ 5.5V/4.95A
    subwf CSACTL,0   ;subtract CSACTL-CSMAX
    btfsc STATUS,CARRY ;check carry for negative result
    goto Sd_inst     ;instant shutdown if below maximum
    nop
    movlw .5        ;current sense absolute minimum @ 5.5V/0.11A
    subwf CSACTL,0   ;subtract CSACTL-CSMIN
    btfss STATUS,CARRY ;check carry for negative result
    goto Sd_inst     ;instant shutdown if below maximum
    nop
    movlw .49       ;bus voltage absolute minimum @ 5.5V/80V
    subwf BV,0       ;subtract BV-BVMIN
    btfss STATUS,CARRY ;check carry for negative result
    goto Sd_inst     ;instant shutdown if below minimum
    nop
    movlw .80       ;bus voltage absolute maximum @ 5.5V/130V
    subwf BV,0       ;subtract BV-BVMAX
    btfsc STATUS,CARRY ;check carry for negative result
    goto Sd_inst     ;instant shutdown if below minimum
    nop
;

;Speed Transition Routine
;
Transit btfss PORTB,SSIN1 ;check operation
    goto Cpl         ;goto next checkpoint
    btfsc PORTB,SSINO ;check operation
    goto Offsw       ;goto off switch
    goto Highsw      ;goto high switch start
Cpl   btfss PORTB,SSIN0 ;check operation
    goto Offsw       ;goto off switch
Lowsw  bcf BITREG,OFFSWTC ;clear off switch flag
    btfss BITREG,HIGHLOW ;check high speed flag
    goto Chk_low     ;continue
    bcf BITREG,HIGHLOW ;clear high/low speed flag
    bsf BITREG,TRANS8 ;set transition flag
    movlw .100        ;number PWMs
    movwf PSTOR0      ;save into PWM Storage LSB
    clrf SECCNTR      ;clear the seconds counter
    goto Timechk     ;goto time check
Chk_low movlw .100        ;load W with SETRPM
    movwf THEO0        ;save into Hall Theoretical LSB
    clrf THEO1        ;clear Hall Theoretical MSB
    bcf STATUS,CARRY  ;clear carry bit
    rrf HSTOR1,0       ;divide Hall storage MSB by 2
    rrf HSTOR0,0       ;divide Hall storage LSB by 2
    sublw .40          ;rotor speed low minimum @ 1200 RPM
    btfsc STATUS,CARRY ;check carry for negative result
    goto Sd_inst      ;instant shutdown if below minimum
    nop
    bcf STATUS,CARRY  ;clear carry bit
    rrf HSTOR1,0       ;divide Hall storage MSB by 2
    rrf HSTOR0,0       ;divide Hall storage LSB by 2
    subiw .133         ;rotor speed low maximum @ 4000 RPM
    btfss STATUS,CARRY ;check carry for negative result
    goto Sd_inst      ;instant shutdown if below minimum
    ;
    goto Timechk     ;goto time check
Highsw bcf BITREG,OFFSWTC ;clear off switch flag
    btfsc BITREG,HIGHLOW ;check high speed flag
    goto Chk_hi       ;continue
    bsf BITREG,HIGHLOW ;set high/low speed flag
    bsf BITREG,TRANS8 ;set transition flag
    moviw .10          ;number PWMs
    movwf PSTOR0      ;save into PWM Storage LSB
    clrf SECCNTR      ;clear the seconds counter
    goto Timechk     ;goto time check
Chk_hi  moviw .160        ;load W with constant SETRPM
    movwf THEO0        ;save into Hall Theoretical LSB
    clrf THEO1        ;clear Hall Theoretical MSB
    bcf STATUS,CARRY  ;clear carry bit
    rrf HSTOR1,0       ;divide Hall storage MSB by 2
    rrf HSTOR0,0       ;divide Hall storage LSB by 2
    subiw .53          ;rotor speed high minimum @ 1600 RPM
    btfsc STATUS,CARRY ;check carry for negative result
    goto Sd_inst      ;instant shutdown if below minimum
    nop
    scf STATUS,CARRY  ;clear carry bit

```

```

        rrf      HSTOR1,J      ;divide Hall storage MSB by 2
        rrf      HSTOR0,0      ;divide Hall storage LSB by 2
        sublw   .166          ;rotor speed high maximum @ 5000 RPM
        btfs s STATUS,CARRY   ;check carry for negative result
        goto    Sd_inst       ;instant shutdown if below minimum
;
        goto    Timechk       ;goto time check
        btfs s BITREG,OFFSWTC ;check if off switch flag already set
        clrf    SECCNTR        ;clear the seconds counter
        bsf     BITREG,OFFSWTC ;set off switch flag
        bsf     BITREG,TRANS8   ;set transition flag
        movlw   .2              ;load 2 seconds into W
        subwf  SECCNTR,0        ;:SECCNTR-2
        btfs s STATUS,I        ;has 2 seconds elapsed?
        goto    Trnsend        ;finished with transition routine
        btfsc  BITREG,HBRAKE   ;check if Hard Brake set
        goto    H_brake        ;hard braking
        goto    Sd_inst       ;instant shutdown
Timechk btfs s BITREG,TRANS8 ;check transition flag
        goto    Trnsend        ;finished with transition routine
        movlw   .8              ;load x seconds into W
        subwf  SECCNTR,0        ;:SECCNTR-x
        btfsc  STATUS,I        ;has x seconds elapsed?
        bcf    BITREG,TRANS8   ;clear transition flag
Trnsend
;
Mainend goto Main           ;perform closed loop operation
;
;
;
;-----+
;-----+
; A/D SUBROUTINES FOR MAIN INTERRUPT
;-----+
;-----+
;A/D Conversion With High/Low Flag Check And Current Averager
;
Ad_main incf  PCYCLE,I      ;increment PWM Cycle Counter
        btfsc  PCYCLE,0      ;check bit 0
        goto   Ad_go         ;start A/D conversion (1:2 PMMs)
        btfsc  PCYCLE,I      ;check bit 1
        goto   Ad_spd        ;setup A/D speed input (1:4 PMMs)
        btfsc  PCYCLE,2      ;check bit 2
        goto   Ad_cur        ;setup A/D bus voltage (1:8 PMMs)
        goto   Ad_bus        ;setup A/D current sense (1:9 PMMs)
;
;Begin A/D, Check Current Sense High, Bus Voltage Low, Shutdown Flag
;
Ad_go  bsf    ADCONO,ADGO   ;set ADGO bit to begin A/D conversion
        return             ;continue
;
;Initialize A/D For Current Sense (AIN0) And Save Speed Input Conversion
;
Ad_cur moviw b'00000001'   ;select fosc/2 and AIN0
        movwf  ADCONO        ;set up A/D
        return             ;continue
;
;Initialize A/D For Bus Voltage (AIN1) And Save Speed Input Conversion
;
Ad_bus moviw b'00001001'   ;select fosc/2 and AIN1
        movwf  ADCONO        ;set up A/D
        return             ;continue
;
;Initialize A/D For Speed Input (AIN2) And Save Current Or Bus Voltage Conversion
;
Ad_spd btfsc PCYCLE,I      ;check bit 2 for current or voltage
        goto   Ad_sous        ;save tid bus voltage
;
;Save Current Sense
;
Ad_sous movwf  ADRES,1      ;load W with last A/D conversion
        movwf  DSACTL        ;save current sense (1:9 PMMs)
        moviw  b'00010001'   ;select fosc/2 and AIN2
        movwf  ADCONO        ;set up A/D and start conversion
        return             ;continue
;
;Save Bus Voltage
;

```

```

Ad_sbus movf    ADRES,0      ;load W with last A/D conversion
           movwf   BV          ;save bus voltage (1:8 PWMS)
           movlw   b'00010001'   ;select fosc/2 and AIN2
           movwf   ADCONO     ;set up A/D
           return      ;continue
;
;
;
;-----;
;      : CHECK ACTUAL VS THEORETICAL
;-----;
;
Timrsub movf    THEO1,0      ;load W with THEO1
           subwf   HSTOR1,0    ;HSTOR1-THEO1
           btfsc   STATUS,2     ;check if MSBs are equal
           goto    Chklsb      ;goto check LSB routine
           btfsc   STATUS,CARRY ;check carry bit
           goto    Speeddn      ;continue
Speedup movf    THEOO,0      ;load THEOO into W
           subwf   HSTOR0,0    ;HSTOR0-THEOO
           movwf   SPDADJ      ;save difference
           goto    Pwmdecr      ;goto decrement routine
Speeddn movf    HSTOR0,0      ;load HSTOR0 into W
           subwf   THEOO,0      ;THEOO-HSTOR0
           movwf   SPDADJ      ;save difference
           goto    Pwmincr      ;goto increment routine
Chklsb movf    THEOO,0      ;load THEOO into W
           subwf   HSTOR0,0    ;HSTOR0-THEOO
           btfsc   STATUS,2     ;check if zero
           return      ;continue
           movwf   SPDADJ      ;HSTOR0-THEOO
           btfss   STATUS,CARRY ;check if positive
           comf    SPDADJ,1    ;complement the difference
           btfsc   STATUS,CARRY ;check if positive
           goto    Pwmincr      ;goto increment routine
Pwmdecr movlw   .7          ;delay loops
           movwf   TEMP2        ;save into temporary counter
           bcf    STATUS,CARRY ;clear carry
           rrf    SPDADJ,1    ;divide by 2
           btfsc   PSTOR0,1    ;decrement PWM storage LSB
           incf    PSTOR0,1    ;check if zero
           incf    PSTOR0,1    ;increment PWM storage MSB
           bcf    STATUS,CARRY ;clear carry
           rrf    SPDADJ,1    ;divide by 2
           btfsc   STATUS,2     ;check if zero
           goto    Tsubena      ;finished with timrsub
           decfsz TEMP2        ;decrement temporary counter
           goto    Pwmdecr      ;continue
           goto    Tsubena      ;finished with timrsub
           movlw   .7          ;delay loops
           movwf   TEMP2        ;save into temporary counter
           bcf    STATUS,CARRY ;clear carry
           rrf    SPDADJ,1    ;divide by 2
           incf    PSTOR0,1    ;increment PWM storage LSB
           btfsc   STATUS,2     ;check if zero
           decf    PSTOR0,1    ;decrement PWM storage MSB
           bcf    STATUS,CARRY ;clear carry
           rrf    SPDADJ,1    ;divide by 2
           btfsc   STATUS,2     ;check if zero
           goto    Tsubena      ;finished with timrsub
           decfsz TEMP2        ;decrement temporary counter
           goto    Pwmdecr      ;continue
Tsubena return      ;end of timrsub routine
;
;
;
;-----;
;      : DISABLE GLOBAL INTERRUPTS
;-----;
;
;Disabling Global Interrupts
;
Gieoff bcf    INTCON,3TE    ;disable global interrupts
           btfsc   INTCON,3TE    ;verify disable

```

```

        goto    Gieoff      ;no...try again
        return   ;return to call
;

;

;

;-----.
;

; HARD BRAKING ROUTINE
;-----.

;

H_brake call    Gieoff      ;disable all interrupts
        bcf    PORTB,ONC      ;set ONC low
        bcf    PORTB,PHA      ;turn phase A off
        bcf    PORTB,PHB      ;turn phase B off
Bphase  clrf    PCNTRO      ;clear PWM counter LSB
        clrf    HCNTRI      ;clear Hall counter MSB
Halloff btfsr  PORTB,HALL    ;alignment of hall signal and phases
        goto   Aphase       ;continue with high hall signal
        nop
        nop
        bsf    PORTB,PHB      ;a delay
        nop
        nop
        bcf    PORTB,PHB      ;turn phase B on
        nop
        nop
        incf   PCNTRO,I      ;for a +1 us pulse
        btfsr  STATUS,Z      ;for a +1 us pulse
        incf   HCNTRI,I      ;check if zero
        movf   HCNTRI,W      ;increment
        sublw .15             ;load W
        btfsr  STATUS,Z      ;compare to timer constant
        goto   Sd_inst       ;motor almost stopped?
        btfsc  STATUS,Z      ;yes...instant shutdown
        goto   Aphase       ;turn phase B on
        nop
        nop
        bcf    PORTB,PHB      ;for a +1 us pulse
        goto   Halloff      ;for a +1 us pulse
        btfsc  STATUS,Z      ;turn phase B off
        goto   Aphase       ;continue pulsing
Aphase  clrf    PCNTRO      ;clear PWM counter LSB
        clrf    HCNTRI      ;clear Hall counter MSB
Hallon  btfsr  PORTB,HALL    ;alignment of hall signal and phases
        goto   Bphase       ;continue with low hall signal
        nop
        bsf    PORTB,PHA      ;a delay
        nop
        nop
        bcf    PORTB,PHA      ;turn phase A on
        nop
        nop
        incf   PCNTRO,I      ;for a +1 us pulse
        btfsr  STATUS,Z      ;for a +1 us pulse
        incf   HCNTRI,I      ;turn phase A off
        movf   HCNTRI,W      ;increment
        sublw .15             ;load W
        btfsr  STATUS,Z      ;compare to timer constant
        goto   Sd_inst       ;motor almost stopped?
        btfsc  STATUS,Z      ;yes...instant shutdown
        goto   Bphase       ;turn phase A on
        nop
        nop
        bcf    PORTB,PHA      ;for a +1 us pulse
        goto   Hallon       ;for a +1 us pulse
        btfsc  STATUS,Z      ;turn phase A off
        goto   Hallon       ;continue loop
;

;

;

;-----.
;

; INSTANT SHUTDOWN ROUTINE
;-----.

;

Sd_inst call    Gieoff      ;disable all interrupts
        bcf    PORTB,ONC      ;set ONC low
        bcf    PORTB,PHA      ;phase A off
        bcf    PORTB,PHB      ;phase B off
        btfsr  PORTB,SSIMO    ;is speed input LSB low
        goto   Sd_inst       ;no...instant shutdown
        btfsr  PORTB,SSIMI    ;is speed input MSB low
        goto   Sd_inst       ;no...instant shutdown
        goto   Start         ;restart microprocessor
;
;
```

```

;-----  

;  

;  TIMER INTERRUPT (1:256 Prescale)  

;  

;-----  

;  

;-----  

; Int_V    movwf   WSAV          ;save W register  

;           movf    STATUS,0      ;load STATUS into W  

;           movwf   STATSAV       ;save STATSAV  

;           btfss   INTCON,TOIF    ;timer interrupt flag?  

;           goto   Pwm_V         ;no...goto PWM interrupt  

;           bcf    INTCON,TOIF    ;reset timer flag  

;           decfsz TCNTR,I       ;decrement timer counter  

;           goto   Int_end       ;finished with interrupt  

;-----  

; End_sec  movf    HCNTRO,0     ;load Hall counter MSB into W  

;           movwf   HSTORI        ;save Hall storage MSB  

;           movf    HCNTRO,0     ;load Hall counter LSB into W  

;           movwf   HSTORO        ;save Hall storage LSB  

;           clrf    HCNTRO        ;clear Hall counter LSB  

;           clrf    HCNTRO        ;clear Hall counter MSB  

;           call   Timrsub        ;call incr/decr routine  

;           movlw   .15            ;set timer to approx 1 sec w/extra 576us  

;           movwf   TCNTR         ;save in timer counter  

;           incf    SECCNTR,I     ;increment elapsed seconds  

;           goto   Int_end       ;finished with interrupt  

;  

;  

;  

;-----  

;  

;  PWM INTERRUPT (25KHz)  

;  

;-----  

;  

;-----  

; Pwm_V    bcf    STATUS,RPO    ;select pg0 registers  

;           bcf    PORTB,PHA      ;turn phase A off  

;           bcf    PORTB,PHB      ;turn phase B off  

;           bcf    INTCCN,INTF    ;clear hall interrupt flag  

;           movf    PSTORO,0      ;move PWM storage LSB into W  

;           movwf   PCNTRO        ;save into PWM counter LSB  

;           incf    HCNTRO,1      ;increment hall counter LSB  

;           btfsc   STATUS,2      ;check for carry  

;           incf    HCNTRO,1      ;increment hall counter MSB  

;           btfsc   PORTB,HALL    ;alignment of hall signal and phases  

;           goto   P_hhi         ;hall high, select phase B high  

;-----  

; P_hlo    bcf    PHASE,4      ;unselect phase B  

;           bsf    PHASE,5        ;select phase A  

;           bsf    STATUS,RPO      ;select pgl registers  

;           bsf    OPTION,INTEDG    ;set rising hall edge detect  

;           bcf    STATUS,RPO      ;select pg0 registers  

;           btfsc   BITREG,HIGHLOW  ;high or low speed?  

;           gotc   P_pulse        ;continue  

;           goto   P_stop         ;continue  

;-----  

; P_hhi    bcf    PHASE,5      ;unselect phase A  

;           bsf    PHASE,4        ;select phase B  

;           bsf    STATUS,RPO      ;select pgl registers  

;           bsf    OPTION,INTEDG    ;set falling hall edge detect  

;           bcf    STATUS,RPO      ;select pg0 registers  

;-----  

; P_pulse  btfss   BITREG,HIGHLOW  ;check high speed flag  

;           movlw   .9             ;delay loops for low (.9)  

;           btfsc   BITREG,HIGHLOW  ;check high speed flag  

;           moviw   .3             ;delay loops for high (.3)  

;           movwf   CNTR2          ;save into general counter  

;-----  

; P_delay  decfsz CNTR2        ;decrement general counter  

;           gotc   P_delay        ;continue delay  

;-----  

; P_mid    decfsz PCNTRO,1      ;decrement general counter  

;           goto   P_pulse        ;continue to PWM  

;-----  

; P_high   movf    PHASE,1      ;load PHASE into W  

;           xorwf   PORTB,1        ;set phase high  

;-----  

; P_stop   moviw   b'00001000'    ;perform XOR on bit 3 of port B  

;           btfsc   BITREG,ONCFLAG  ;check for OWC  

;           xorwf   PORTB,1        ;switch OWC high/low  

;-----  

; Int_end  movf    STATSAV,0     ;load STATUS into W  

;           movwf   STATUS          ;restore STATUS register  

;           movf    WSAV,0          ;load WSAV into W  

;           retfie                ;end of interrupt  

;  

;-----  


```

```

; DESCRIPTION This program is designed to implement a constant CFM based on the comparison
; of a torque calculation and an RPM-torque look-up table. High/Low Switch
; determines the selectable speed ranges for furnace.
;

;
;

; -----
; REGISTER/BIT ADDRESSES
; -----
;

;Predefined Register Addresses
;

LIST P=16C71           ;Identifies PIC16C71

;

LSB    equ    0h          ;Division constants
B0     equ    0h
B1     equ    1h
B2     equ    2h
B3     equ    3h
PIC71  equ    0h          ;16C71 Special-purpose registers
INDF   equ    0h
RTCC   equ    1h
OPTION equ    01h
PCL    equ    2h
STATUS equ    3h
FSR    equ    4h
PORTA  equ    5h
PORTB  equ    6h
PORTC  equ    7h
ADCON0 equ    8h
ADCON1 equ    08h
ADRES  equ    9h
PCLATH equ    0Ah
INTCON equ    0Bh
TRISA  equ    05h
TRISB  equ    06h
IRP    equ    7h          ;16C71 STATUS bits
RP1    equ    5h
RPO    equ    5h
TO     equ    4h
PD     equ    3h
Z      equ    2h
DC     equ    1h
CARRY  equ    0h
GIE    equ    7h          ;16C71 INTCON register bits
ADIE   equ    6h
TOIE   equ    5h
INTE   equ    4h
RBIE   equ    3h
TOIF   equ    2h
INTF   equ    1h
RBIF   equ    0h
ADCS1  equ    7h          ;16C71 ADCN0 register bits
ADCS0  equ    6h
CHSI   equ    4h
CHSO   equ    3h
ADGO   equ    2h
ADIF   equ    1h
ADOH   equ    0h
PCFG1  equ    1h          ;16C71 ADCN1 register bits
PCFG0  equ    0h
INTEDG equ    0h          ;16C71 OPTION register bits
;

;PORTA register bits
;

SELECT3 equ    1h          ;Selection Bit 3
SELECT2 equ    1h          ;Selection Bit 2
;

;PORTC register bits
;

```

```

HALL    equ    0h          ;Hall Signal Input
SSIN1   equ    1h          ;Set Speed Input MSB
SSINO   equ    2h          ;Set Speed Input LSB
OWC     equ    3h          ;Operational Window Correct
PHB     equ    4h          ;Phase B Output
PHA     equ    5h          ;Phase A Output
SELECT0  equ    6h          ;Selection Bit 0
SELECT1  equ    7h          ;Selection Bit 1
;
;BITREG register bits
;
OFFSWTC equ    0h          ;Off Switch Flag
TRANSB   equ    1h          ;Eight Second Transition Flag
HIGHLOW  equ    2h          ;High/Low Speed Flag
WAITOWC  equ    3h          ;Wait For OWC Flag
OWCFLAG  equ    6h          ;OWC Flag
SFTSTRT  equ    7h          ;Soft Start Flag
;
;BITREG2 register bits
;
SLOWFST equ    0h          ;Slow Or Fast Transition Flag
FOURSEC  equ    1h          ;Four Seconds Start Flag
CALCFRQ  equ    3h          ;Calculation Frequency Flag
STARTUP  equ    5h          ;Starting 15 Seconds Flag
;
;Define General Purpose Registers
;
H_BYTE   equ    0Ch         ;Calculation MSB
L_BYTE   equ    0Dh         ;Calculation LSB
CNTR1   equ    07h         ;General Counter 1
CNTR2   equ    0Fh         ;General Counter 2
PCNTRO  equ    10h         ;PWM Counter LSB
PSTORO  equ    11h         ;PWM Storage LSB
PCYCLE  equ    12h         ;PWM Cycle Counter
TCITR   equ    13h         ;Timer Counter
SECCNTR equ    14h         ;Seconds Counter
HCNTR1  equ    15h         ;Hall Edges Counter MSB
HCNTR0  equ    16h         ;Hall Edges Counter LSB
HSTOR1  equ    17h         ;Hall Edges Storage MSB
HSTOR0  equ    18h         ;Hall Edges Storage LSB
TCALC   equ    19h         ;Torque Calculation
TTHEO   equ    1Ah         ;Torque Constant Theoretical
BV      equ    1Bh         ;Bus Voltage Value
CSSUM1  equ    1Ch         ;Current Sense Sum MSB
CSSUM0  equ    1Dh         ;Current Sense Sum LSB
CSACTL  equ    1Eh         ;Current Sense Actual Value
BITREG  equ    1Fh         ;Bit Register For Loop Control
BITREG2 equ    20h         ;Bit Register For Loop Control C
PHASE   equ    21h         ;Select Phase A Or B
WSAV    equ    22h         ;Save Value Of W Register
STATSAV equ    23h         ;Save Value Of Status Register
MULCND equ    24h         ;8-bit Multiplicand
MULPLR  equ    25h         ;8-bit Multiplier
TEMP    equ    26h         ;Temporary Register
BLKFLUE equ    27h         ;Current Calculation for Blocked Flue
;
;
;-----;
;
; CODE ORIGINS
;
;-----;
;
;Establish Program Origin And Interrupt Vectors
;
        org    00h           ;reset vector
;
        org    04h           ;interrupt vector
;
        org    07h           ;start of code
;
;
;
;-----;
;
; INITIALIZE REGISTERS
;
```

```

;
;-----;
;Initialize Page 1 Registers
;
Start  bsf    STATUS,RPO      ;select pg1 registers
        movlw  b'00011111'   ;set port A I/O (1=input)
        movwf  TRISA          ;send values to port A
        movlw  b'11000111'   ;set port B I/O (1=input)
        movwf  TRISB          ;send values to port B
        movlw  .2              ;set AIN0,1 to analog; AIN2,3,4 to digital
        movwf  ADCON1          ;set ADCON1 register
        movlw  b'01000111'   ;select prescaler, timer and INT edges
        movwf  OPTION          ;set OPTION register
;
;Clear Variables
;
        bcf    STATUS,RPO      ;select pg0 registers
        clrf   PORTA          ;clear
        clrf   PORTB          ;clear
        clrf   ADCONO          ;clear
        clrf   ADRES           ;clear
        clrf   PCLATH          ;clear
        clrf   INTCON          ;clear
        clrf   PCYCLE          ;clear
        clrf   PHASE           ;clear
        clrf   WSAV            ;clear
        clrf   STATSAV         ;clear
        clrf   SECCNTR         ;clear
        clrf   BITREG          ;clear
        clrf   BITREG2         ;clear
;
;Initialize Variables
;
        movlw  .15             ;set timer to approximately 1 sec
        movwf  TCNTR          ;timer counter
        movlw  .75             ;set starting value
        movwf  BV               ;save bus voltage
        movwf  CSACTL         ;save current sense
;
;
;
;-----;
;START UP ROUTINE
;
;-----;
;Check Selection Bits And Low/High Fine Commands
;
Highlow :btfs  PORTA,SELECT0  ;check selection bit 0
;        goto  Start          ;return to start up
;        btfs  PORTA,SELECT1  ;check selection bit 1
;        goto  Start          ;return to start up
;        btfs  PORTA,SELECT2  ;check selection bit 2
;        goto  Start          ;return to start up
;        btfs  PORTA,SELECT3  ;check selection bit 3
;        goto  Start          ;return to start up
;        btfs  PORTB,SSINO    ;wait for speed input LSB to go low
;        goto  Go              ;continue
;        btfs  PORTB,SSIN1    ;wait for speed input MSB to go high
;        goto  Start          ;restart micro
;
;Perform Soft Start With Increasing Torque
;
Go     bsf    BITREG,SFTSTRT ;set soft start flag
        movlw  .191            ;number of hall edges
        movwf  TEMPF           ;save into general counter
;
Haitchg :movf  TEMP.C          ;load W
        movwf  PSTORO          ;save increasing torque delay counter
        movlw  .5              ;shift right loops
        movwf  CNTR1           ;save into general counter
;
Shifting :bcf   STATUS,CARRY  ;clear carry bit
        rrf   PSTORO,1          ;divide by 2
        decfsz CNTR1           ;decrement general counter
        goto  Shifting         ;continue shifting
        movf  PSTORO,1          ;load W
        btfs  STATUS,0          ;check if zero
        goto  Msetup            ;continue with Main Setup

```

```

        movlw .2          ;number of hall edges doubled
        movwf L_BYTE      ;save into temporary counter
        btfsc PORTB,HALL ;alignment of hall signal and phases
        goto Hihall       ;continue with high hall signal
        bcf PORTB,PHB    ;turn phase B off
        bcf PHASE,5      ;select phase A
        bcf PHASE,4      ;unselect phase B
        bcf STATUS,RPO   ;select pgl registers
        bcf OPTION,INTEDG;set rising hall edge detect
        bcf STATUS,RPO   ;select pg0 registers
        goto Quar_0       ;continue
Hihall  bcf PORTB,PHA  ;turn phase A off
        bcf PHASE,5      ;unselect phase A
        bcf PHASE,4      ;select phase B
        bcf STATUS,RPO   ;select pgl registers
        bcf OPTION,INTEDG;set falling hall edge detect
        bcf STATUS,RPO   ;select pg0 registers
Quar_0  movf TEMP,0     ;number of pwms (y of x*y)
        movwf PCNTRO     ;save into general counter
Quar_1  movlw .25       ;number of pwms (x of x*y)
        movwi H_BYTE      ;save into general counter
        btfsc PORTB,SSINO ;is speed input LSB low
        goto Hal_1        ;continue
        btfss PORTB,SSIN1 ;is speed input MSB high
        goto Sd_inst      ;no...instant shutdown
        movf PHASE,0      ;load PHASE into W
        xorwf PORTB,i    ;set phase high
        movf PSTORO,0    ;load W
        sublw .6          ;6-PSTORO delay loops
        movwf CNTR1       ;save into general counter
        decfsz CNTR1     ;decrement general counter
        goto Del_0        ;continue delay
        movf PHASE,0      ;load PHASE into W
        xorwf PORTB,i    ;set phase low
        btfsc INTCON,INTF ;check if hall edge received
        goto Hallcnd      ;continue
        movf PSTORO,0    ;load W
        movwf CNTR1       ;save into general counter
        decfsz CNTR1     ;decrement general counter
        goto Del_1        ;continue delay
Del_1   decfsz H_BYTE    ;decrement general counter
        goto Hal_1        ;continue pulsing
        decfsz PCNTRO     ;decrement general counter
        goto Quar_1       ;continue loop
        movf PHASE,0      ;load PHASE into W
        xorwf PORTB,i    ;set phase high
        btfsc PORTB,SSINO ;is speed input LSB low
        goto Edg_1         ;continue
        btfss PORTB,SSIN1 ;is speed input MSB high
        goto Sd_inst      ;no...instant shutdown
Edg_1   btfss INTCON,INTF ;check if hall edge received
        goto Edgwait      ;continue waiting for hall edge
Hallcnd bcf INTCON,INTF ;clear hall edge if present
        decfsz L_BYTE     ;decrement general counter
        goto Halchq2      ;continue loop
        decfsz TEMP       ;decrement general counter
        goto Halcnq       ;continue loop
;
;Setup For PWM, Timer, And Main Program Processing
;
Msetup  bcf PORTB,PHA  ;turn phase A off
        bcf PORTB,PHB  ;turn phase B off
        clrf HCNTRO    ;clear
        clrf HCNTRI    ;clear
        clrf HSTORO    ;clear
        clrf HSTOR1    ;clear
        clrf SECCNTR    ;clear seconds counter
        moviw .55       ;PWM starting point LSB
        movwf PSTORO    ;save in PWM storage LSB
        movwf PCNTRO    ;save into PWM counter LSB
        moviw .15       ;set timer to approximately 1 sec
        movwi TCNTR     ;timer counter
        moviw b'00110000';clear flags and enable TMRO and PWM
        movwi INTCON    ;write INTCON register
        bcf INTCON,GIE  ;global enable
        bcf INTCON,INTF ;clear hall edge if present soft start exit
        bci BITREG,ONCFLG ;set CWC flag
        bci BITREG,WAITONC;set WAITONC flag
        bci BITREG,HIGHLOW;set HIGHLOW flag

```

```

;
;
;
;-----  

; MAIN PROGRAM  

;-----  

;  

Main  

;  

;Call A/D Routine  

;  

Ad_call call Ad_main ;A/D conversion routine  

;  

;Check For Full Restriction On Start Up Before Flame  

;  

fullrst btfsc BITREG2,STARTUP ;check if starting 15 seconds expired
    goto Calctrq ;continue
    movf SECCNTR,0 ;load W
    sublw .15 ;15 seconds to stabilize
    btfsc STATUS,CARRY ;check if positive
    goto Calctrq ;continue
    movf SECCNTR,0 ;load W
    subiw .17 ;2 second check (24 total secs before flame)
    btfsc STATUS,Z ;check if zero
    bsf BITREG2,STARTUP ;set starting seconds flag
    moviw .4 ;load W
    movwf BLKFLUE ;save into current
    movf BV,0 ;load with bus voltage actual
    movwf MULCND ;save into multiplicand
    moviw .92 ;load constant
    movwf MULPLR ;save into multiplier
    call Mult98c ;BV*constant
    bcf STATUS,CARRY ;clear carry
    rlf L_BYTE,0 ;multiply by 2
    rlf H_BYTE,0 ;multiply by 2
    addwf BLKFLUE,1 ;add to current
    movf HSTOR1,0 ;check MSB
    btfss STATUS,Z ;check flag
    call Owcoff ;turn OWC off
    movf BLKFLUE,0 ;estimated start up current with blocked flue
    subwf CSACTL,0 ;subtract CSACTL-CSCALC
    btfss STATUS,CARRY ;check carry for negative result
    call Owcoff ;turn OWC off
;  

;Calculate Torque Constant
;  

Calctrq btfsc BITREG,TRANS8 ;check flag
    goto Transit ;continue with transition
    btfss BITREG2,CALCFRQ ;check flag
    goto Transit ;continue with transition
    bcf BITREG2,CALCFRQ ;clear flag
    bcf STATUS,CARRY ;clear carry bit
    movf HSTOR0,0 ;load Hall storage LSB
    movwf MULCND ;save into multiplicand
    movf CSACTL,0 ;load Current Sense
    movwf MULFLR ;save into multiplier
    Mulpoly call Mult98c ;HSTOR0 * CSACTL
    movf HSTOR1,1 ;check HSTOR1
    btfsc STATUS,Z ;check if zero
    goto Incres ;continue
    movf CSACTL,0 ;load into W
    addwf H_BYTE,1 ;add to H_BYTE
    Incres btfsc BITREG,HIGHLOW ;check high or low speed
    goto Vadjust ;continue
    bcf STATUS,CARRY ;clear carry
    rlf L_BYTE,1 ;multiply by 2
    rlf H_BYTE,1 ;multiply by 2
    Vadjust movf BV,0 ;load with bus voltage actual
    subiw .112 ;subtract x-BV with voltage x 1 120VAC
    btfsc STATUS,Z ;check if zero
    goto Calcdns ;finished with calculation
    btfss STATUS,CARRY ;check if negative
    goto Adjup ;goto adjust up
    Adtdown movwf TEMP ;save W temporarily
    bcf STATUS,CARRY ;clear carry
    fdiv TEMP,1 ;divide by 1
    bcf STATUS,CARRY ;clear carry

```

```

rrf    TEMP,I      ;divide by 2
movf   TEMP,0      ;load W
subwf  H_BYTEx,1   ;subtract from torque value
bcf    STATUS,CARRY ;clear carry
rrf    TEMP,0      ;divide by 2
subwf  H_BYTEx,1   ;subtract from torque value
goto   Calcend    ;goto end of calculation
Adjup
        movwf  TEMP      ;save W temporarily
        comf   TEMP,I    ;take complement of temporary
        bcf    STATUS,CARRY ;clear carry
        rrf    TEMP,I    ;divide by 2
        bcf    STATUS,CARRY ;clear carry
        rrf    TEMP,I    ;divide by 2
        movf   TEMP,0      ;load W
        addwf  H_BYTEx,1   ;add to torque value
        bcf    STATUS,CARRY ;clear carry
        rrf    TEMP,0      ;divide by 2
        addwf  H_BYTEx,1   ;add to torque value
        movf   H_BYTEx,0   ;load W
        movwf  TCALC     ;save torque calculated value
;

;Speed Transition Routine
;
Transit
        btfss  PORTB,SSIN1  ;check operation
        goto   Cpl         ;goto next checkpoint
        btfsc  PORTB,SSINO  ;check operation
        gotc   Offsw       ;goto off switch
        goto   Highsw      ;goto high switch start
Cpl
        btfss  PORTB,SSINO  ;check operation
        goto   Offsw       ;goto off switch
Lowsw
        bcf    BITREG,OFFSWTC ;clear off switch flag
        btfss  BITREG,HIGHLCW ;check high speed flag
        goto   Lowspd      ;continue
        bcf    BITREG,HIGHLOW ;clear high/low speed flag
        baf    BITREG,TRANSB  ;set transition flag
        bsf    BITREG,WAITOWC ;set waitowc flag
        bcf    BITREG2,SLOWFST ;clear slow/fast flag
        movlw  .55          ;number PWMs
        movwf  PSTOR0      ;save into PWM Storage LSB
        clrf   SECCNTR     ;clear the seconds counter
        goto   Timechk     ;continue
Lowspd
        btfsc  BITREG,TRANSB ;check if in transition
        goto   Timechk     ;continue
        bcf    STATUS,CARRY ;clear carry bit
        rrf    HSTOR1,0      ;divide Hall storage MSB by 2
        rrf    HSTOR0,0      ;divide Hall storage LSB by 2
        sublw .60           ;low minimum 60 @ 1800 RPM
        btfsc  STATUS,CARRY ;check carry for negative result
        call   Owoff        ;turn OWC off
        bcf    STATUS,CARRY ;clear carry bit
        rrf    HSTOR1,0      ;divide Hall storage MSB by 2
        rrf    HSTOR0,0      ;divide Hall storage LSB by 2
        sublw .108          ;low maximum 108 @ 3200 RPM
        btfss  STATUS,CARRY ;check carry for negative result
        call   Owoff        ;turn OWC off
Ls_cabl
        movlw  .230          ;load maximum current value (# pts + offset-1)
        subwf  HSTOR0,0      ;subtract maximum from HSTOR0
        btfss  STATUS,CARRY ;check for overflow
        gotc   Table        ;continue
        movlw  .110          ;load maximum table value (# pts-1)
        goto   Tab_ls       ;goto call low speed table
Table
        movlw  .120          ;load offset into W (# starting pts omitted)
        subwf  HSTOR0,0      ;subtract offset from HSTOR0
        btfss  STATUS,CARRY ;check for overflow
        movlw  .0             ;load minimum table value
        btfsc  STATUS,CARRY ;check for overflow
        movlw  .0             ;load minimum table value
;      movf   HSTOR0,0      ;if no offset, uncomment this line
;      call   Tableis      ;call the low speed table
Tab_ls
        moviw .1
        addlw .1
        movwf TTHEO        ;save torque theoretical
        moviw .2
        subwf TTHEC,:      ;TTHEC,:
        gotc   Timechk      ;continue
Highsw
        bcf    BITREG,OFFSWTC ;clear off switch flag
        btfsc  BITREG,HIGHLCW ;check high speed flag
        gotc   Highspd      ;continue
        bcf    BITREG,HIGHLOW ;set high/low speed flag

```

```

bsf    BITREG.TRANS8 ;set transition flag
bsf    BITREG.WAITOWC ;set waitowc flag
bcf    BITREG2.SLOWFST ;clear slow/fast flag
bsf    BITREG2.FOURSEC ;set four seconds flag
movlw .20 ;number PWMs
movwf PSTOR0 ;save into PWM Storage LSB
clrf SECCNTR ;clear the seconds counter
goto Timechk ;continue
Highspd btfsc BITREG.TRANS8 ;check if in transition
goto Timechk ;continue
bcf STATUS,CARRY ;clear carry bit
rrf HSTOR1,0 ;divide Hall storage MSB by 2
rrf HSTOR0,0 ;divide Hall storage LSB by 2
sublw .93 ;high minimum 93 @ 2800 RPM
btfs STATUS,CARRY ;check carry for negative result
call Owoff ;turn OWC off
bcf STATUS,CARRY ;clear carry bit
rrf HSTOR1,0 ;divide Hall storage MSB by 2
rrf HSTOR0,0 ;divide Hall storage LSB by 2
sublw .145 ;high maximum 145 @ 4300 RPM
btfs STATUS,CARRY ;check carry for negative result
call Owoff ;turn OWC off
Hs_tabl movf HSTOR1,0 ;check if MSB set
btfs STATUS,I ;check if zero
goto Over ;over 255
movlw .180 ;value to decrement
subwf HSTOR0,0 ;HSTOR0-180
btfs STATUS,CARRY ;check if negative
movlw .0 ;clear W
goto Tab_hs ;goto tab'
Over movlw .76 ;value to increment
addwf HSTOR0,0 ;HSTOR0+76
Tab_hs call Tablehs ;call the high speed table
;
; addlw .1
; movwf TTHEO ;save torque theoretical
; movlw .3
; subwf TTHEO,1
; goto Timechk ;continue
Offsw btfs BITREG.OFFSWTC ;check if off switch flag already set
clrf SECCNTR ;clear the seconds counter
bsf BITREG.OFFSWTC ;set off switch flag
bsf BITREG.TRANS8 ;set transition flag
movlw .2 ;load x seconds into W
subwf SECCNTR,0 ;SECCNTR-x
btfs STATUS,I ;has x seconds elapsed?
goto Sd_inst ;instant shutdown
goto Trnsend ;finished with transition routine
Timechk btfs BITREG.TRANS8 ;check transition flag
goto Trnsend ;finished with transition routine
movlw .8 ;load x seconds into W
subwf SECCNTR,0 ;SECCNTR-x
btfs STATUS,I ;has x seconds elapsed?
goto Trnsend ;continue
bcf BITREG.TRANS8 ;clear transition flag
Trnsend goto Main ;perform closed loop operation
;
;
;
-----
;
; A/D SUBROUTINES FOR MAIN INTERRUPT
;
-----
;
;A/D Conversion With High/Low Flag Check And Current Average
;
Ad_main ctfscc PCYCLE,0 ;check bit 0
getc Ad_go ;start A/D conversion :1:2 PWMs
btfscc PCYCLE,1 ;check bit 1
getc Ad_cur ;setup A/D current sense :1:4 PWMs
getc Ad_bus ;setup A/D bus voltage :1:4 PWMs
Ad_cnts incfscc PCYCLE,1 ;increment PWM cycle counter
return ;finished A/D routine
;_avgr bcf STATUS,CARRY ;clear carry bit
;_if CSSUM0,:1 ;divide by 1
;_if CSSUM1,:1 ;divide by 2
; bcf STATUS,CARRY ;clear carry bit
;_if CSSUM0,:1 ;divide by 1

```

```

rlf    CSSUM1,1      :divide by 2
bcf    STATUS,CARRY   :clear carry bit
movf   CSSUM1,0      :load into W
movwf  CSACTL        :save current sense
clrf   CSSUM0        :clear current sense sum LSB
clrf   CSSUM1        :clear current sense sum MSB
Ad_end return         :finished A/D routine

;
;Begin A/D, Check Current Sense High & Bus Voltage Low
;

Ad_go  bsf    ADCONO,ADGO   :set ADGO bit to begin A/D conversion
       movlw .223          :current sense relative maximum @ 5.45V/4.75A
       subwf CSACTL,0       :subtract CSACTL-CSHIGH
       btfsc STATUS,CARRY   :check carry for negative result
       call  Owcoff         :turn OWC off
       movlw .63            :bus voltage relative minimum @ 5.45V/81V
       subwf BV,0           :subtract BV-BVLOW
       btfss STATUS,CARRY   :check carry for negative result
       call  Owcoff         :turn OWC off
       goto  Ad_cont        :return to main

;
;Initialize A/D For Current Sense (AIN0) And Save Bus Voltage
;

Ad_cur  movf   ADRES,0      :load W with last A/D conversion
       movwf  BV             :save bus voltage (1:4 PWMs)
       movlw  b'00000001'     :select fosc/2 and AIN0
       movwf  ADCONO        :set up A/D
       goto  Ad_cont        :return to main

;
;Initialize A/D For Bus Voltage (AIN1) And Save Current Sense
;

Ad_bus  movf   ADRES,0      :load W with last A/D conversion
       addwf  CSSUM0,i       :add to sum (1:4 PWMs)
       btfsc STATUS,CARRY   :check for carry
       incf   CSSUM1,1       :increment sum MSB
       movlw  b'00001001'     :select fosc/2 and AIN1
       movwf  ADCONO        :set up A/D and start conversion
       goto  Ad_cont        :return to main

;
;
;
-----
; TIMER INCREMENT/DECREMENT AND SET OWC AND SLOW/FAST FLAG
;
-----
;

Timrsub btfsc  BITREG1,SLOWFST :check transition speed
       goto   Chkflgs        :continue
       movlw .3                :frequency of incr/decr
       ancfw SECCNTR,0         :and with counter
       btfss STATUS,1           :check if zero
       return
Chkflgs btfsc  BITREG1,TRANS8 :check if in transition
       return
       btfss BITREG1,OWCFLAG  :check if OWC
       return
       btfss BITREG1,STARTUP   :check if starting 15 seconds expired
       return
Chkdifff movf   TCALC,0       :load into W
       subwf TTHEO,0          :TTHEO-TCALC
       btfsc STATUS,0           :check if zero
       return
       btfsc STATUS,CARRY     :check if positive
       goto   Pwmdecr         :goto decrement routine
Pwmincr incf   PSTOR0,1       :increment PWM storage LSB
       return
Pwmdecr decf   PSTOR0,1       :decrement PWM storage LSB
       return
       incf   PSTOR0,1           :increment PWM storage LSB
Tsubend return          :end of Time_id routine

;
;
;
-----
; DISABLE LOCAL INTERRUPTS
;
-----

```

```

;
;Disabling Global Interrupts
;
Gieoff bcf    INTCON,GIE      ;disable global interrupts
        btfsc  INTCON,GIE      ;verify disable
        goto   Gieoff          ;no...try again
        return             ;return to call
;
;
;
;-----
;
;  8x8 CODE EFFICIENT MULTIPLIER
;
;-----
;
Mult88c clrf   H_BYTE
        clrf   L_BYTE
        moviw  .8
        movwf  CNTR1
        movf   MULCND,0
        bcf    STATUS,CARRY
M_loop rrf    MULPLR
        btfsc  STATUS,CARRY
        addwf  H_BYTE,1
        rrf    H_BYTE,1
        rrf    L_BYTE,1
        decfsz CNTR1
        goto   M_loop
        retl   0
;
;
;
;-----
;
;  OWC OFF ROUTINE
;
;-----
;
Owoff bcf    BITREG,WAITOWC ;clear flag
        bcf    BITREG,OWCFLAG ;clear flag
        bcf    PORTB,OWC       ;set OWC signal low
        return             ;return
;
;
;
;-----
;
;  INSTANT SHUTDOWN ROUTINE
;
;-----
;
Sd_inst call   Gieoff          ;disable all interrupts
        bcf    PORTB,OWC       ;set OWC low
        bcf    PORTB,PHA       ;phase A off
        bcf    PORTB,PHB       ;phase B off
        btfsc  PORTE,SSINO    ;is speed input LSB low
        goto   Sd_inst         ;no...instant shutdown
        btfsc  PORTB,SSINI    ;is speed input MSB low
        goto   Sd_inst         ;no...instant shutdown
        goto   Start           ;restart microprocessor
;
;
;
;-----
;
;  TIMER INTERRUPT (1:128 Prescale Or 65536us)
;
;-----
;
Int_1'  movwf  NSAV          ;save W register
        movf   STATUS,0          ;load STATUS into W
        movwf  STATSAV          ;save STATSAV
        ctfs  INTCON,T0IF      ;timer interrupt flag?
        goto   PWM_V            ;no...goto PWM interrupt
        bcf    INTCON,T0IF      ;reset timer flag
        osf   INTREG,CALCFREQ  ;set flag
        decfsz T0INTVAL         ;decrement timer counter
        goto   Int_end          ;finished with interrupt

```

```

End_sec    movf    HCNTRO,0      ;load Hall counter MSB into W
           movwf    HSTOR1      ;save Hall storage MSB
           movf    HCNTRO,0      ;load Hall counter LSB into W
           movwf    HSTOR0      ;save Hall storage LSB
           clrf    HCNTRO      ;clear Hall counter LSB
           clrf    HCNTRO      ;clear Hall counter MSB
           call    Timrsub      ;call Timrsub
           movlw   .15          ;set timer to approx 1 sec
           movwf    TCNTR       ;save in timer counter
           incfsz  SECCNTR,1    ;increment elapsed seconds
           goto   Int_end      ;finished with interrupt
           btfsc   BITREG2,FOURSEC ;check four seconds flag
           bsf     BITREG2,SLOWFST ;set flag after approximately nine seconds
           bsf     BITREG2,FOURSEC ;set flag after 4 1/2 seconds
Tintend    goto   Int_end      ;finished with interrupt
;
;
;
;-----;
;-----;
; PWM INTERRUPT (25KHz)
;-----;
;
;-----;
Pwm_`    bcf    STATUS,RPO      ;select pg0 registers
           bcf    PORTB,PHA      ;turn phase A off
           bcf    PORTB,PHB      ;turn phase B off
           bcf    INTCON,INTF    ;clear hall interrupt flag
           movf    PSTOR0,0      ;move PWM storage LSB into W
           movwf    PCNTRO      ;save into PWM counter LSB
           incf    HCNTRO,1      ;increment hall counter LSB
           btfsc   STATUS,I      ;check for carry
           incf    HCNTRO,1      ;increment hall counter MSB
           btfsc   PORTB,HALL    ;alignment of hall signal and phases
           goto   P_hhi        ;hall high, select phase B high
P_hlo     bcf    PHASE,4      ;unselect phase B
           bsf    PHASE,5      ;select phase A
           baf    STATUS,RPO      ;select pgl registers
           baf    OPTION,INTEDG  ;set rising hall edge detect
           bcf    STATUS,RPO      ;select pg0 registers
           goto   P_pulse      ;continue
P_hhi     bcf    PHASE,5      ;unselect phase A
           bsf    PHASE,4      ;select phase B
           baf    STATUS,RPO      ;select pgl registers
           baf    OPTION,INTEDG  ;set falling hall edge detect
           bcf    STATUS,RPO      ;select pg0 registers
           goto   P_pulse      ;select high speed flag
           btfss   BITREG,HIGHLOW ;delay loops for low
           movlw   .8          ;check high speed flag
           btfsc   BITREG,HIGHLOW ;delay loops for high
           movlw   .3          ;delay loops for high
           movwf    CNTR2      ;save into general counter
P_dlay    decfsz  CNTR2      ;decrement general counter
           goto   P_dlay      ;continue delay
P_mid     decfsz  PCNTRO,1    ;decrement general counter
           goto   P_pulse      ;continue to PWM
P_high    movf    PHASE,0      ;load PHASE into W
           xorwf    PORTB,1      ;set phase high
P_stop    moviw   b'00001000'  ;perform XOR on bit 3 of port B
           btfsc   BITREG,OWCFLAG ;check for OWC
           xorwf    PORTB,1      ;switch OWC high/low
Int_end   movf    STATSAV,0    ;load STATUS into W
           movwf    STATUS       ;restore STATUS register
           movf    WSAV,0        ;load WSAV into W
           retfie
;
;
;
;-----;
;-----;
; LOW SPEED LOOK UP TABLE
;-----;
;
;-----;
;
org    200h
;
Tableis  movwf    TEMP
           movlw   high_low_tbl
           movwf    PCLATH
           movlw   low_low_tbl

```

```
addwf  TEMP,0
btfac  STATUS,CARRY
incf  PCLATH,1
movwf  PCL
Low_tbl
retlw .20
retlw .20
retlw .21
retlw .21
retlw .21
retlw .22
retlw .22
retlw .22
retlw .22
retlw .22
retlw .23
retlw .23
retlw .23
retlw .23
retlw .23
retlw .24
retlw .24
retlw .24
retlw .25
retlw .25
retlw .25
retlw .26
retlw .26
retlw .26
retlw .27
retlw .27
retlw .28
retlw .28
retlw .28
retlw .29
retlw .29
retlw .30
retlw .30
retlw .31
retlw .31
retlw .32
retlw .32
retlw .33
retlw .33
retlw .34
retlw .34
retlw .35
retlw .36
retlw .36
retlw .37
retlw .37
retlw .38
retlw .39
retlw .39
retlw .40
retlw .41
retlw .41
retlw .42
retlw .43
retlw .43
retlw .44
retlw .45
retlw .46
retlw .46
retlw .47
retlw .48
retlw .49
retlw .49
retlw .50
retlw .51
retlw .52
retlw .53
retlw .53
retlw .54
retlw .55
retlw .56
retlw .57
retlw .58
retlw .59
retlw .59
retlw .60
```

```

retlw .61
retlw .62
retlw .63
retlw .64
retlw .65
retlw .66
retlw .67
retlw .68
retlw .69
retlw .70
retlw .71
retlw .72
retlw .73
retlw .74
retlw .75
retlw .76
retlw .77
retlw .78
retlw .80
retlw .81
retlw .82
retlw .83
retlw .84
retlw .85
retlw .86
retlw .88
retlw .89
retlw .90
retlw .91
retlw .92
retlw .94
retlw .95
retlw .96
retlw .97
retlw .99
retlw .100
retlw .101
;
;
;
-----
; HIGH SPEED LOOK UP TABLE
;-----
;
Tables mowwf TEMP
    movlw high Hi_tbl
    movwf PCLATH
    movlw low Hi_tbl
    addwf TEMP,0
    btfsc STATUS,CARRY
    incf PCLATH,1
    mowwf PCL
Hi_tbl
    retlw .31
    retlw .31
    retlw .32
    retlw .32
    retlw .32
    retlw .32
    retlw .33
    retlw .33
    retlw .33
    retlw .33
    retlw .34
    retlw .34
    retlw .34
    retlw .35
    retlw .35
    retlw .35
    retlw .35
    retlw .35
    retlw .36
    retlw .36
    retlw .37
    retlw .37
    retlw .37
    retlw .38
    retlw .38

```

retlw .39  
retlw .39  
retlw .39  
retlw .40  
retlw .40  
retlw .41  
retlw .41  
retlw .42  
retlw .42  
retlw .43  
retlw .43  
retlw .44  
retlw .44  
retlw .45  
retlw .46  
retlw .46  
retlw .47  
retlw .47  
retlw .48  
retlw .48  
retlw .49  
retlw .50  
retlw .50  
retlw .51  
retlw .52  
retlw .52  
retlw .53  
retlw .54  
retlw .54  
retlw .55  
retlw .56  
retlw .57  
retlw .57  
retlw .58  
retlw .59  
retlw .60  
retlw .60  
retlw .61  
retlw .62  
retlw .63  
retlw .64  
retlw .64  
retlw .65  
retlw .66  
retlw .67  
retlw .68  
retlw .69  
retlw .70  
retlw .70  
retlw .71  
retlw .72  
retlw .73  
retlw .74  
retlw .75  
retlw .76  
retlw .77  
retlw .78  
retlw .79  
retlw .80  
retlw .91  
retlw .92  
retlw .83  
retlw .84  
retlw .85  
retlw .86  
retlw .87  
retlw .88  
retlw .89  
retlw .91  
retlw .92  
retlw .93  
retlw .94  
retlw .95  
retlw .96  
retlw .97  
retlw .99  
retlw .100  
retlw .101  
retlw .102

```
retlw .103
retlw .105
retlw .106
retlw .107
retlw .108
retlw .110
retlw .111
retlw .112

end
```

1   What is claimed is:

2   1. A brushless dc motor assembly comprising:

3           a brushless dc motor;

4           a control board having at least one output connected to a  
5        stator winding of said brushless dc motor for providing current  
6        flow to said stator winding;

7            controll electronics on said control board for controlling  
8        said current flow to said stator winding based on desired motor  
9        operating characteristics, said control electronics including a  
10      housekeeping power supply for providing a stable 5v DC signal from  
11      a rectified AC line voltage.

12

13   2. A brushless dc motor assembly comprising:

14           a brushless dc motor;

15           a control board having at least one output connected to a  
16        stator winding of said brushless dc motor for providing current  
17        flow to said stator winding;

18           controll electronics on said control board for controlling  
19        said current flow to said stator winding based on desired motor  
20        operating characteristics, said control electronics including a  
21      MOSFET output amplifier having a power zener diode connected to the  
22      drain thereof, said power zenor dissipating temporary back emf  
23      resulting from switching of said MOSFET from an on to an off state.

24

25   3. A brushless dc motor assembly comprising:

26           a brushless dc motor;

1           a hall device mounted to a stator of said brushless dc motor,  
2        said hall device providing a signal representative of the  
3        rotational speed of a rotor of said motor to a control board,

4           said control board having at least one output connected to a  
5        stator winding of said brushless dc motor for providing current  
6        flow to said stator winding; and

7           control electronics on said control board for controlling said  
8        current flow to said stator winding responsive to said signal.

9

10     4. A brushless dc motor assembly comprising:

11        a brushless dc motor;

12        a control board mounted to a heatsink, said control board  
13        having at least one output connected to a stator winding of said  
14        brushless dc motor for providing current flow to said stator  
15        winding;

16        control electronics comprising a MOSFET on said control board  
17        for controlling said current flow to said stator winding based on  
18        desired motor operating characteristics,

19        wherein said heat sink is attached to said MOSFET for  
20        dissipating heat generated by said MOSFET.

21

22     5. A brushless dc motor assembly comprising:

23        a brushless dc motor;

24        a control board having at least one output connected to a  
25        stator winding of said brushless dc motor for providing current  
26        flow to said stator winding;

1        controll electronics on said control board for controlling  
2        said current flow to said stator winding based on desired motor  
3        operating characteristics, said control electronics including  
4        microprocessor for controlling said output according to desired  
5        operating characteristics.

6

7        6. A brushless dc motor according to claim 5, wherein  
8        microprocessor comprises a set of programmed instructions for  
9        pulsing said output to slowly ramp up the speed of a rotor of said  
10      motor.

11

12      7. A brushless dc motor according to claim 5, wherein said motor  
13      is a 2-phase motor and said microprocessor controls a phase a  
14      output and a phase b output to drive electronics using programmed  
15      instructions based on desired operating specifications.

16

17      8. A microprocessor for controlling a phase a and a phase b stator  
18      winding of a brushless dc motor, said microprocessor comprising  
19      programmed instructions for controlling dc output to said windings  
20      based on user defined specifications.

21

22      9. A brushless dc motor according to claim 5, wherein  
23      microprocessor comprises a set of programmed instructions for  
24      controlling the pulse width of said output.

25

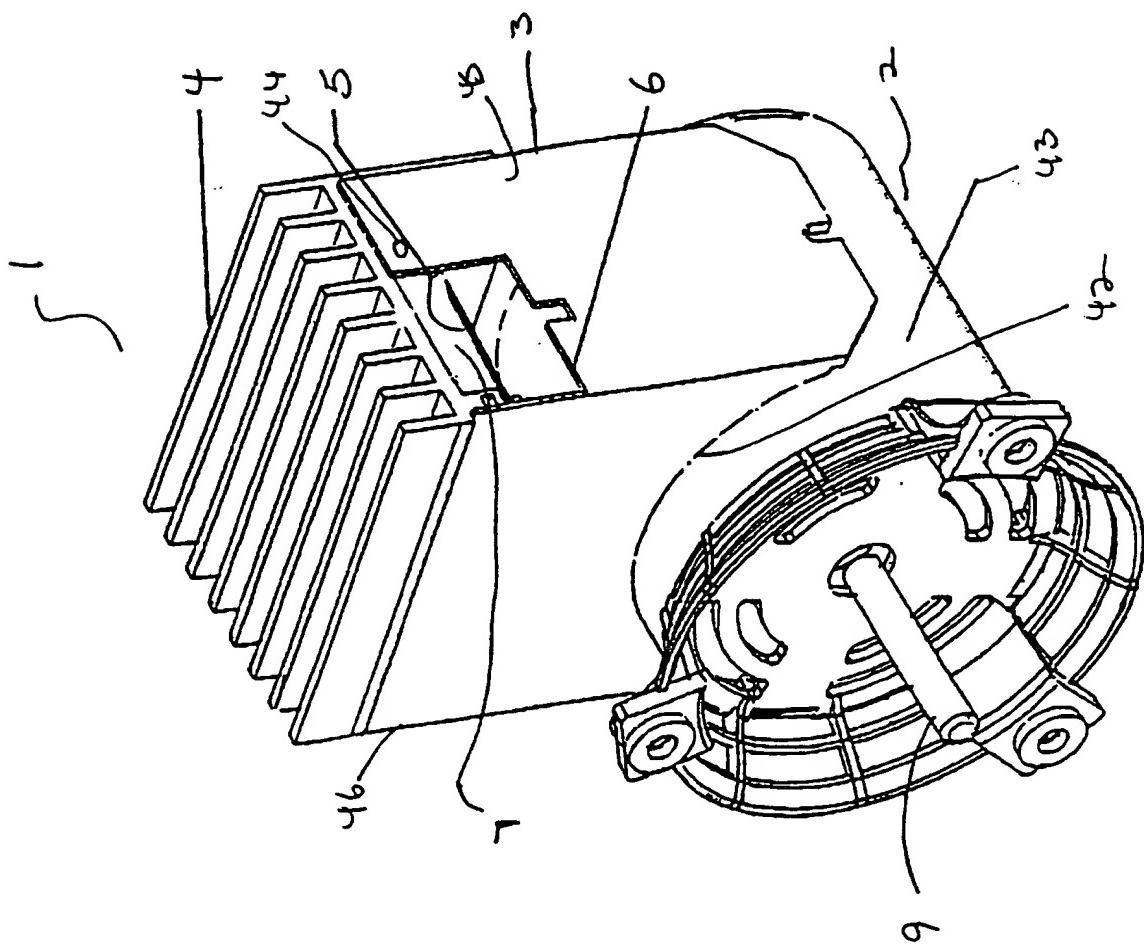


FIG. 1

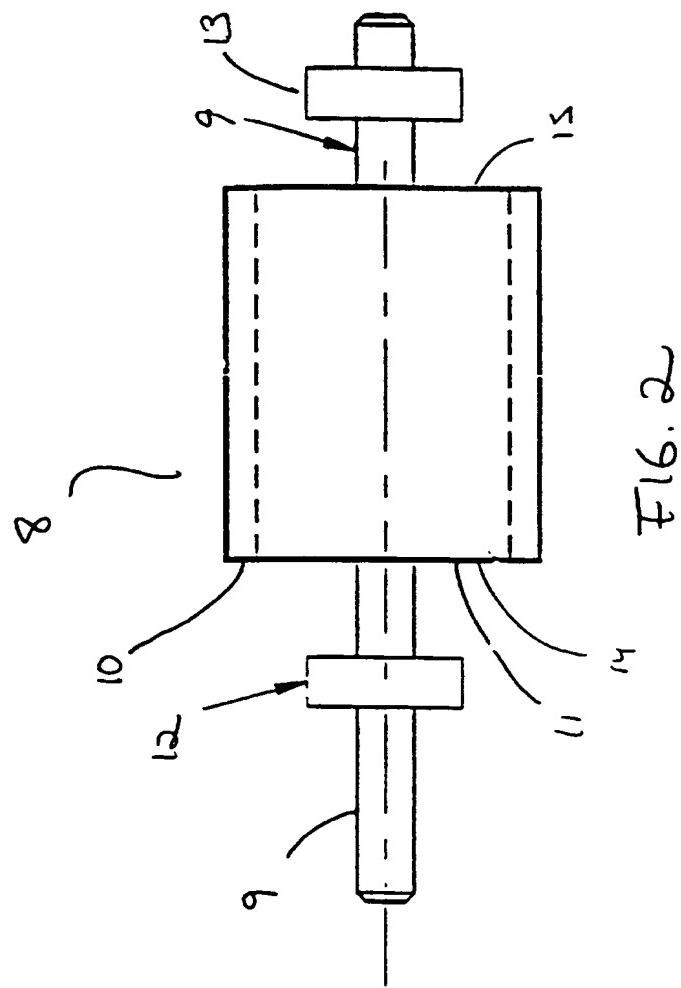


FIG. 2

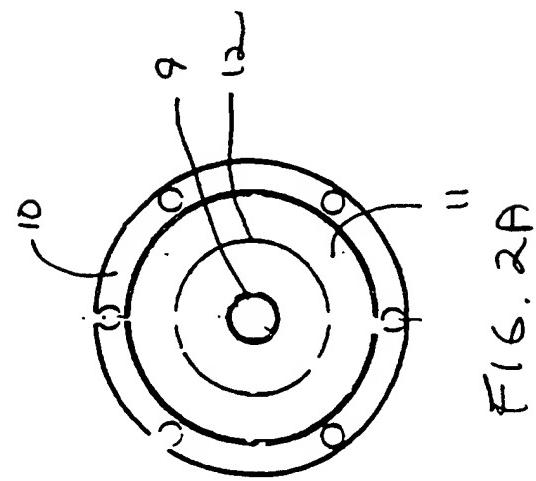
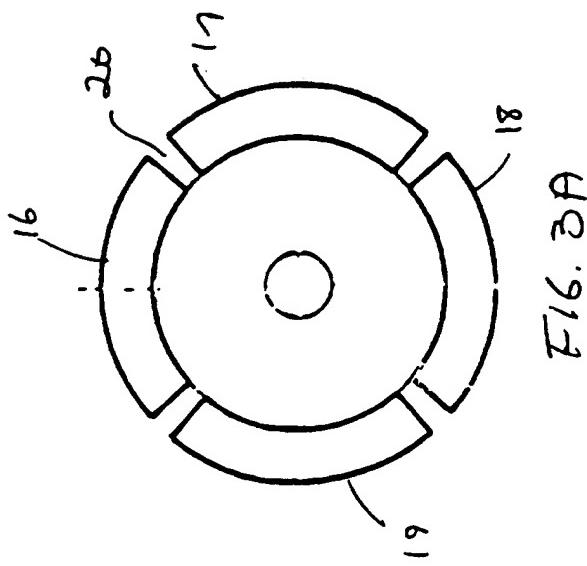
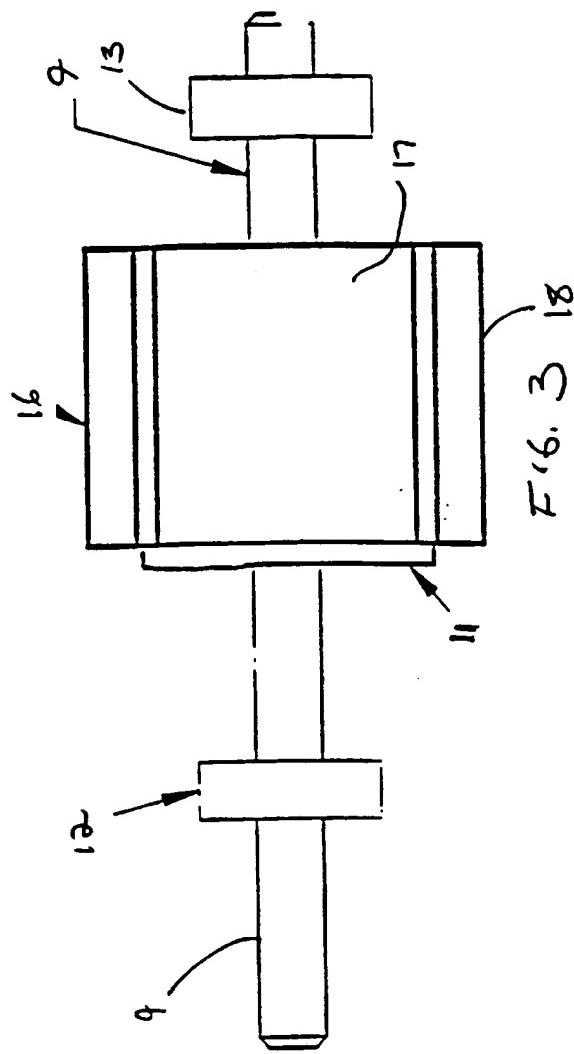


FIG. 2A



4/28

FIG. 4

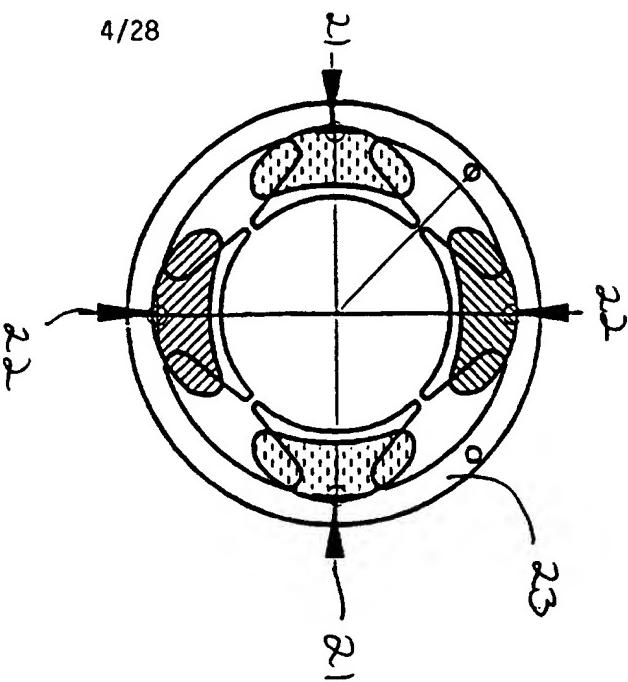
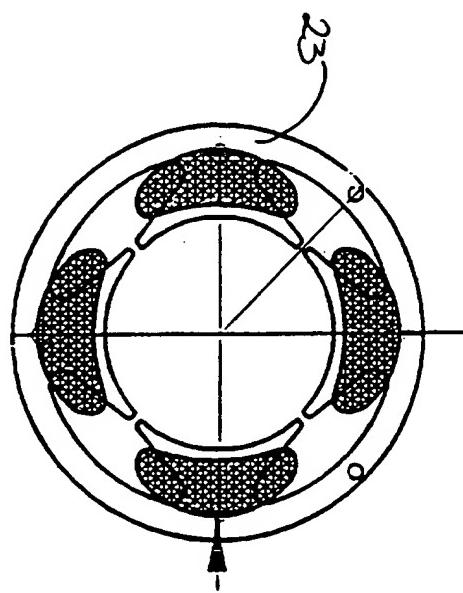
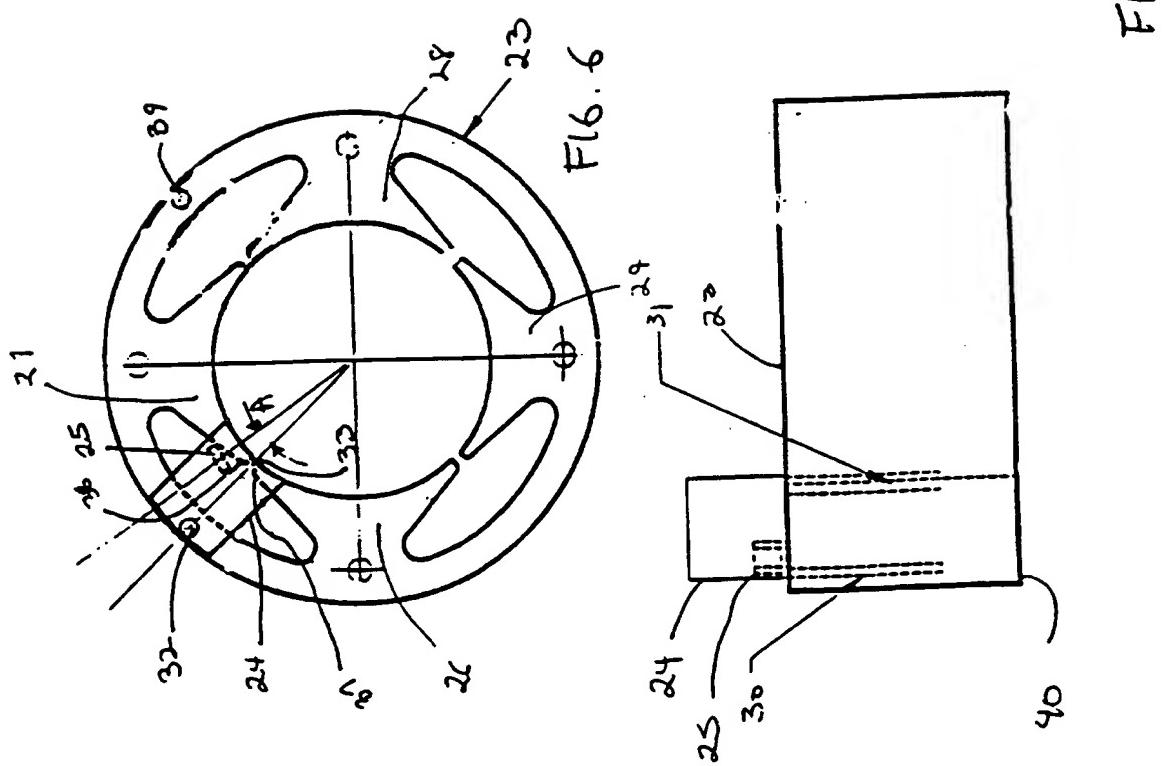


FIG. 5



5/28



6/28

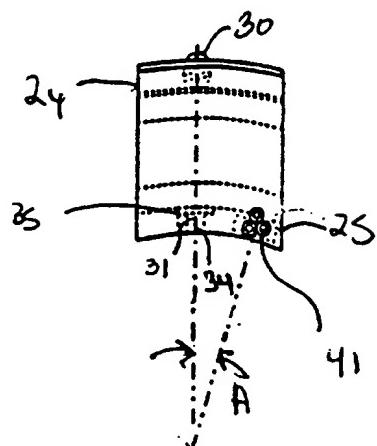


FIG. 8C

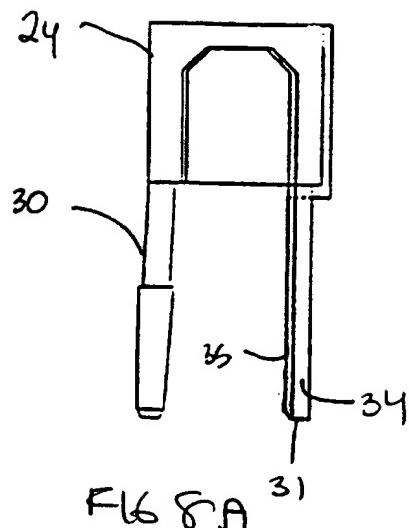


FIG. 8A

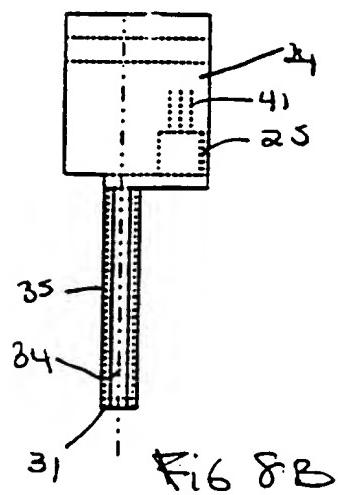


FIG. 8B

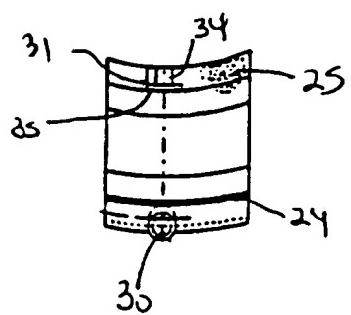
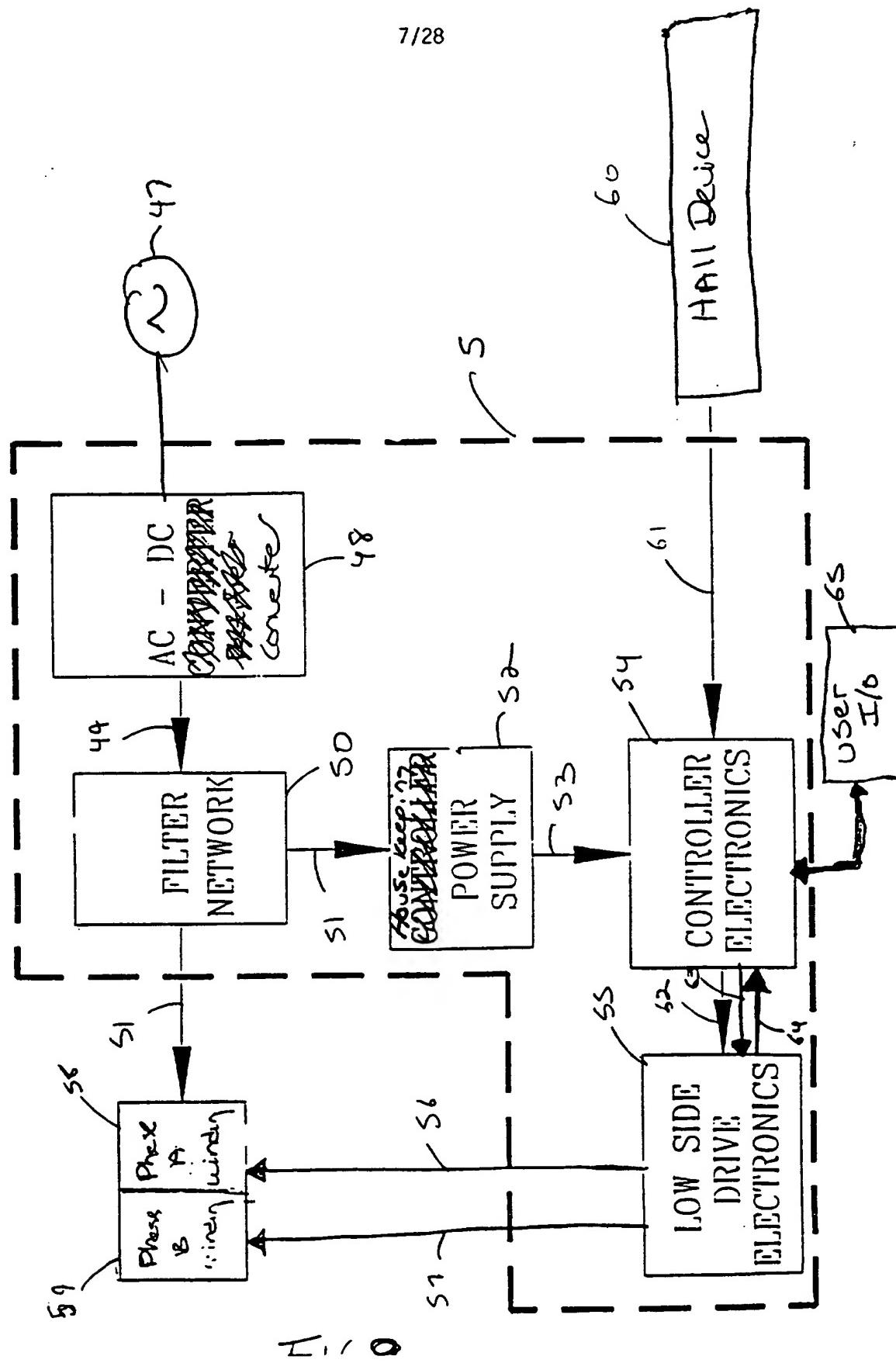
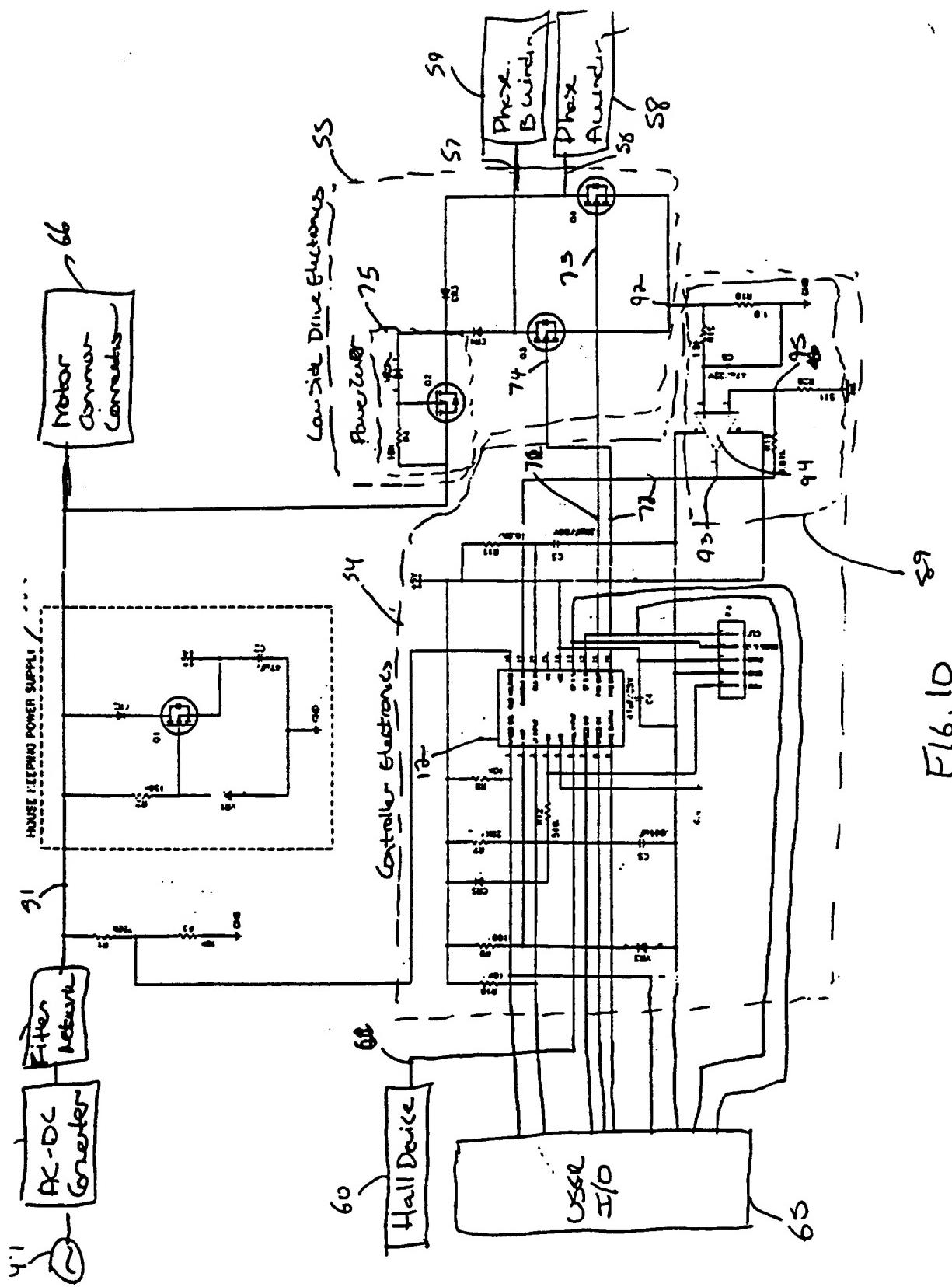


FIG. 8D





9/28

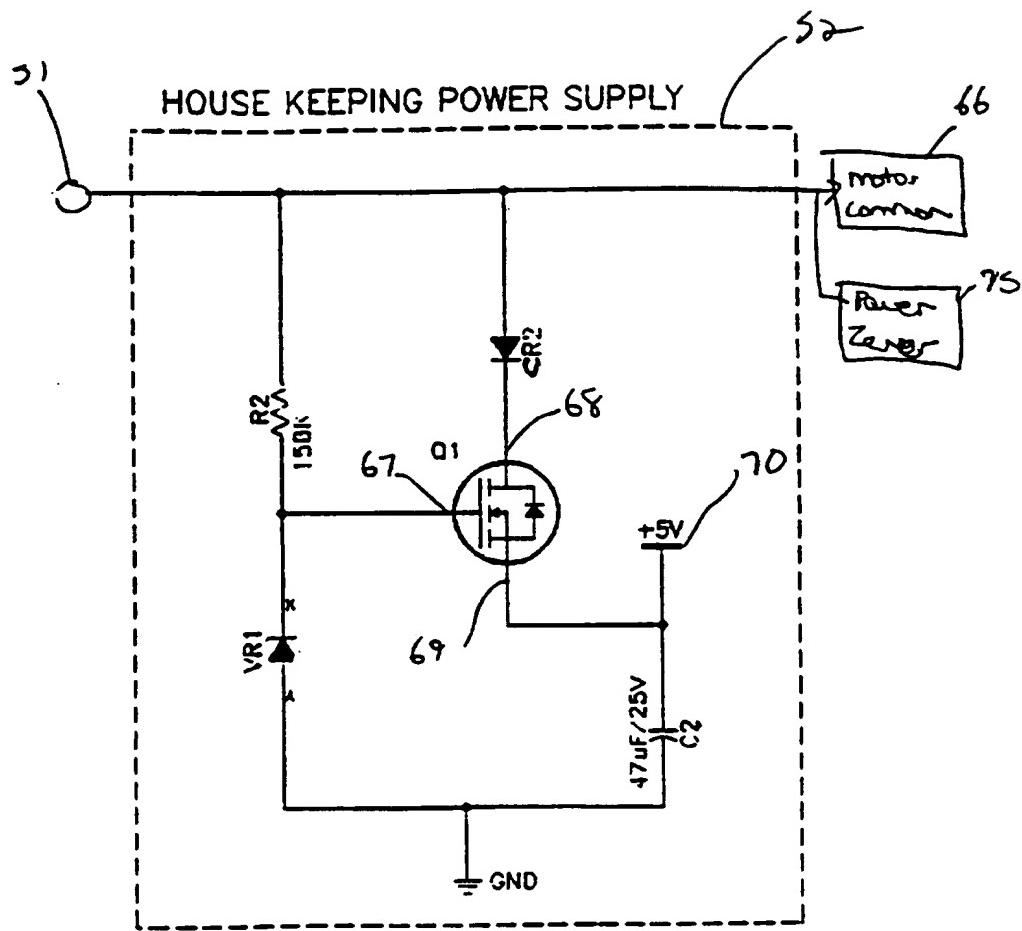


FIG. 11

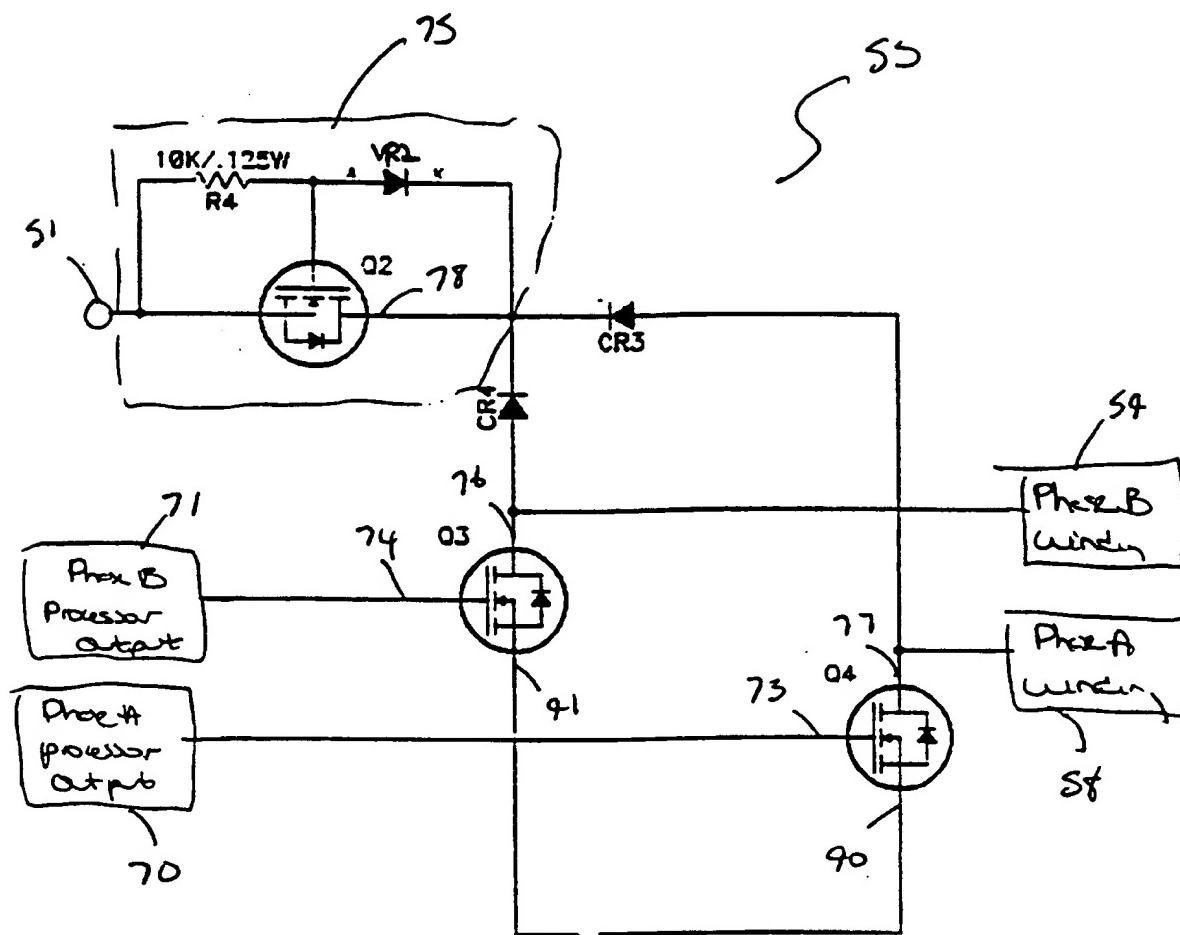


Fig. 12

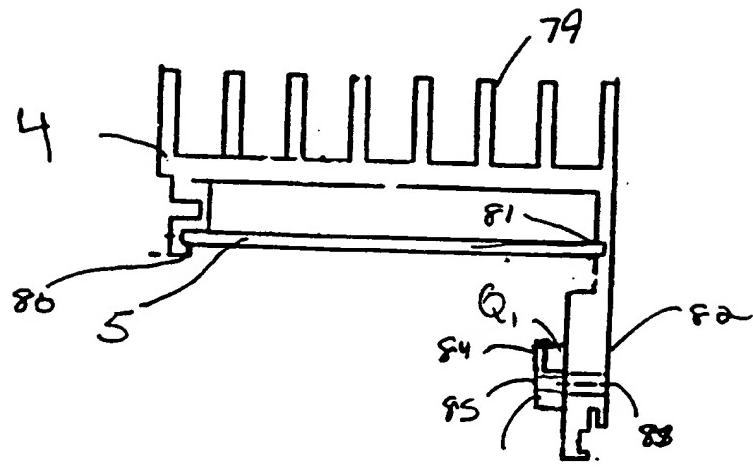


FIG. 16.13

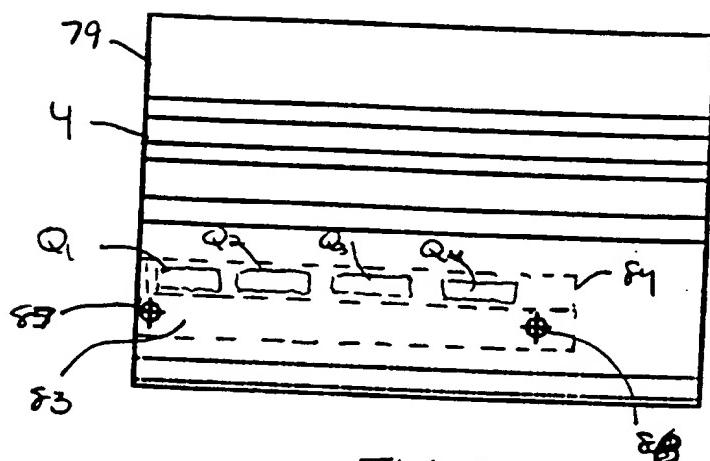


FIG. 16.14

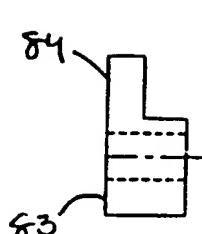


FIG. 16.16

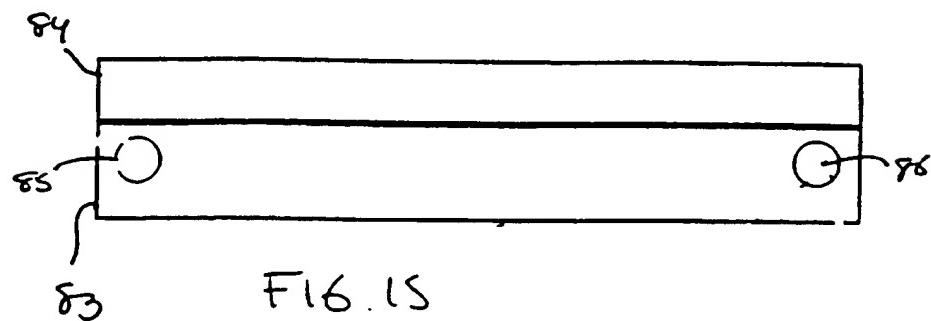


FIG. 16.15

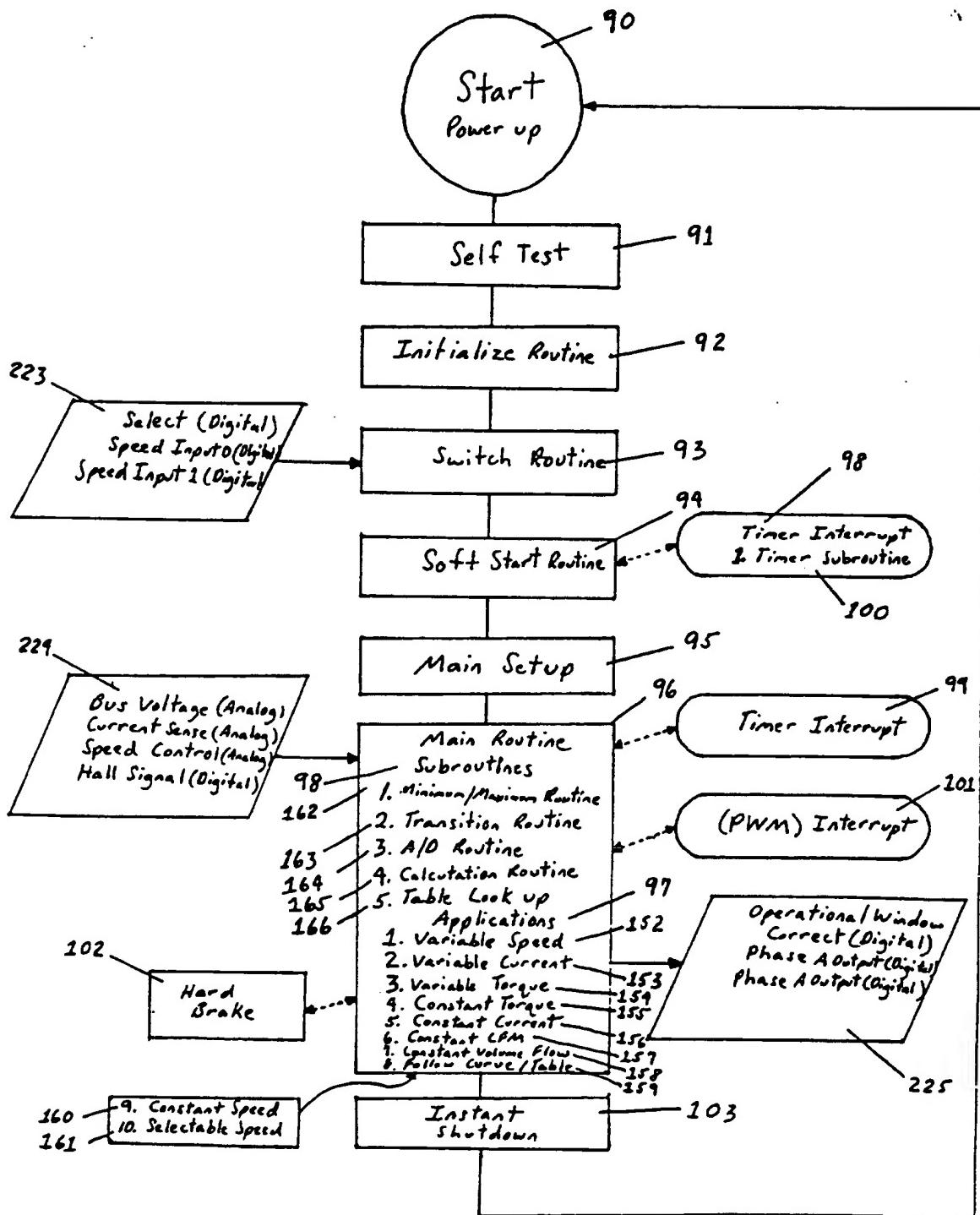


FIG.17

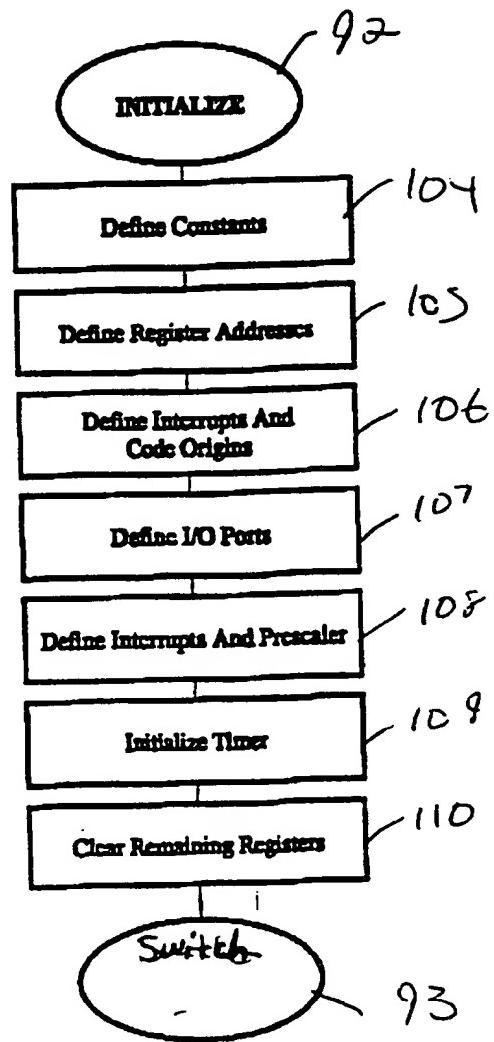


FIG. 18

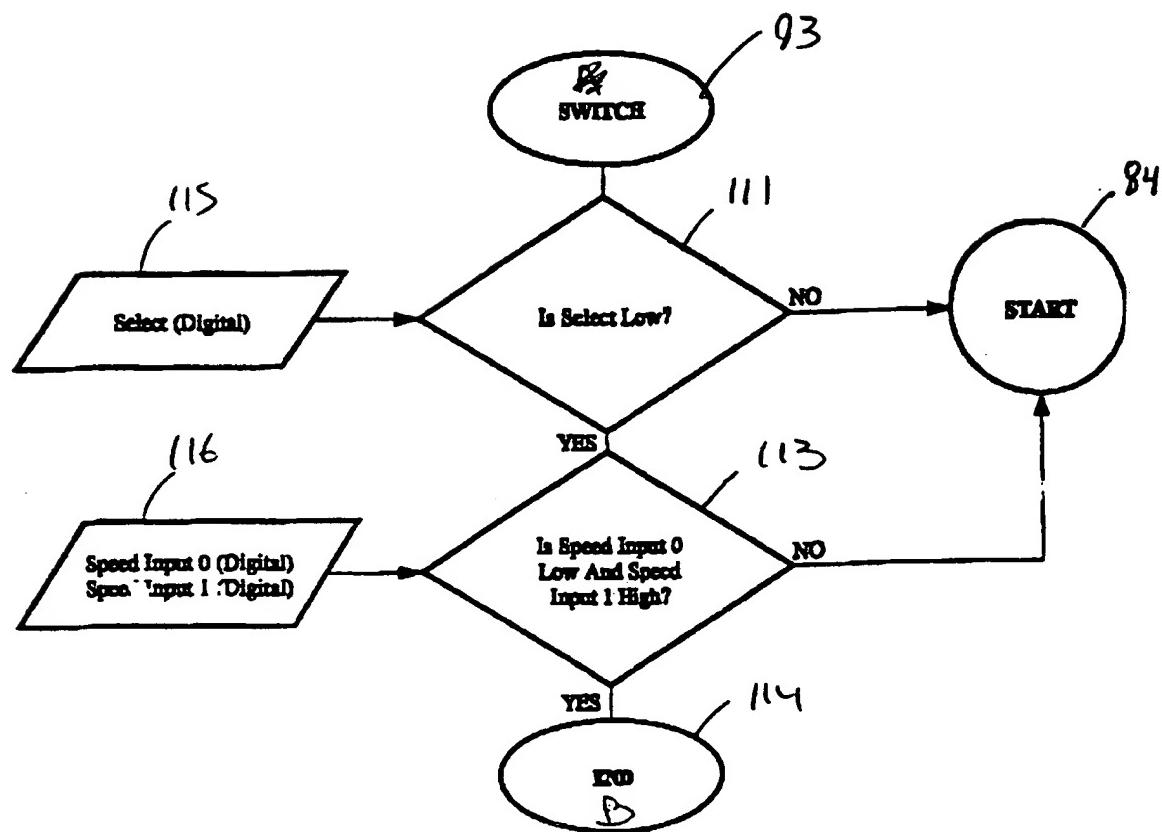
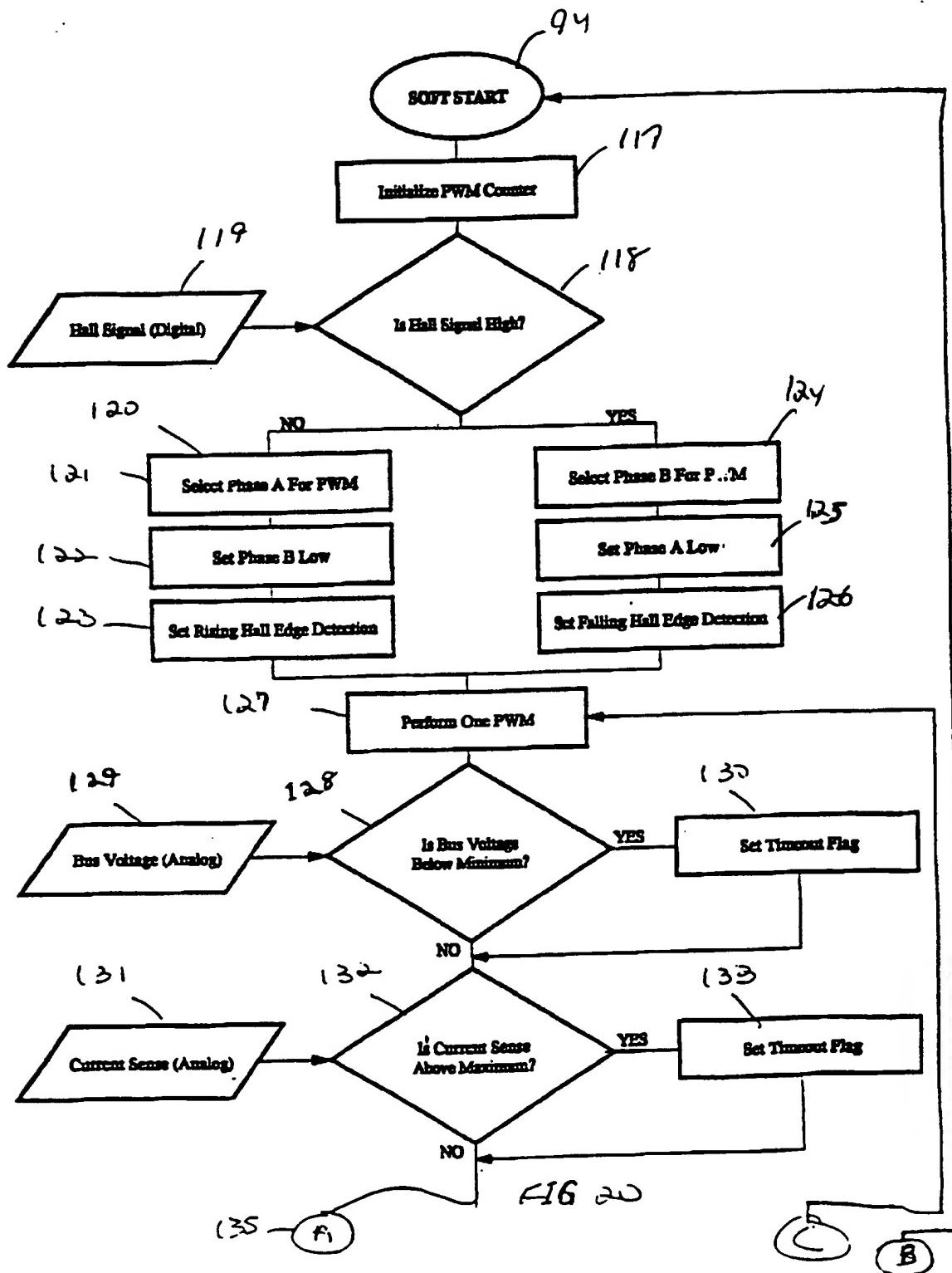
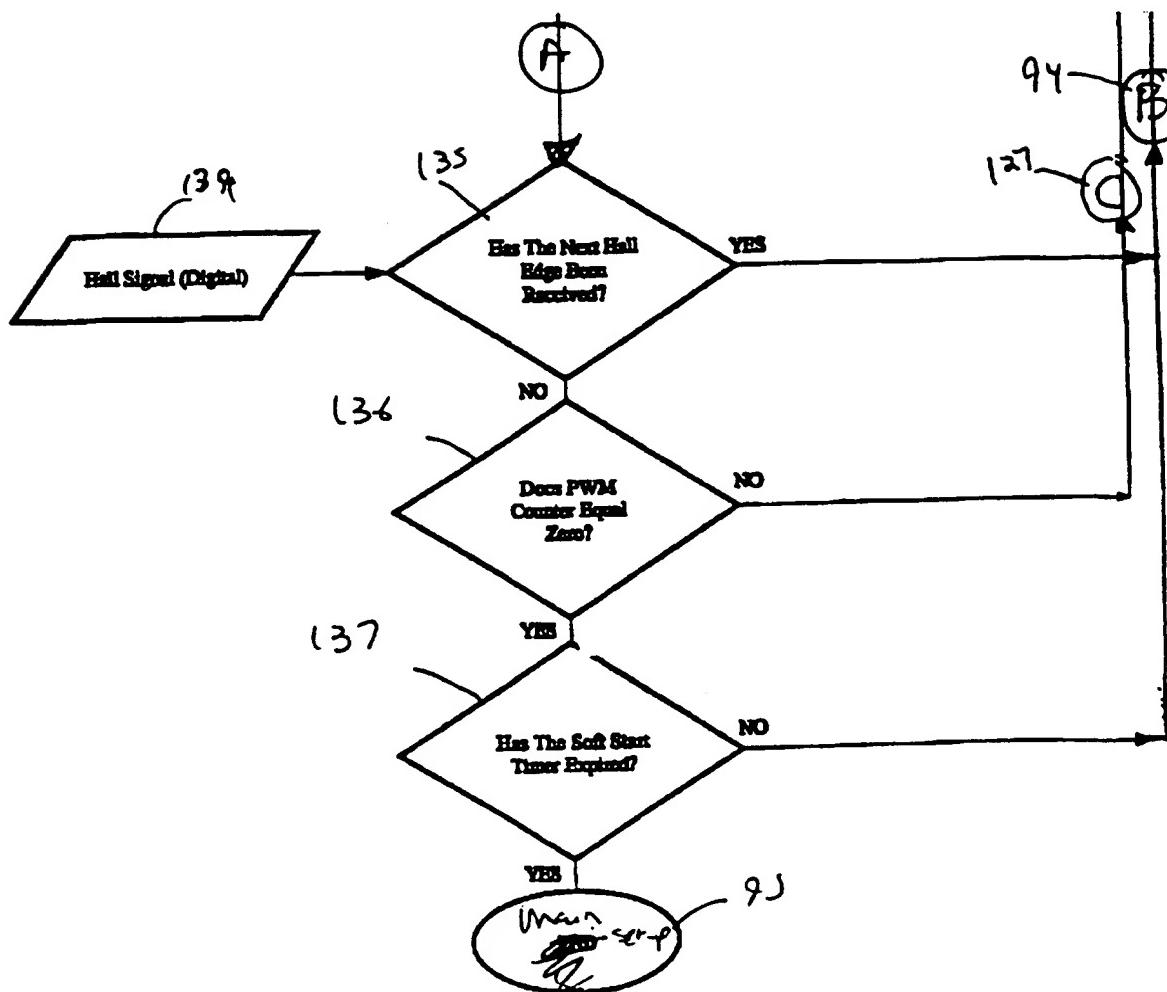


FIG. 19



16/28



F16. 24

17/28

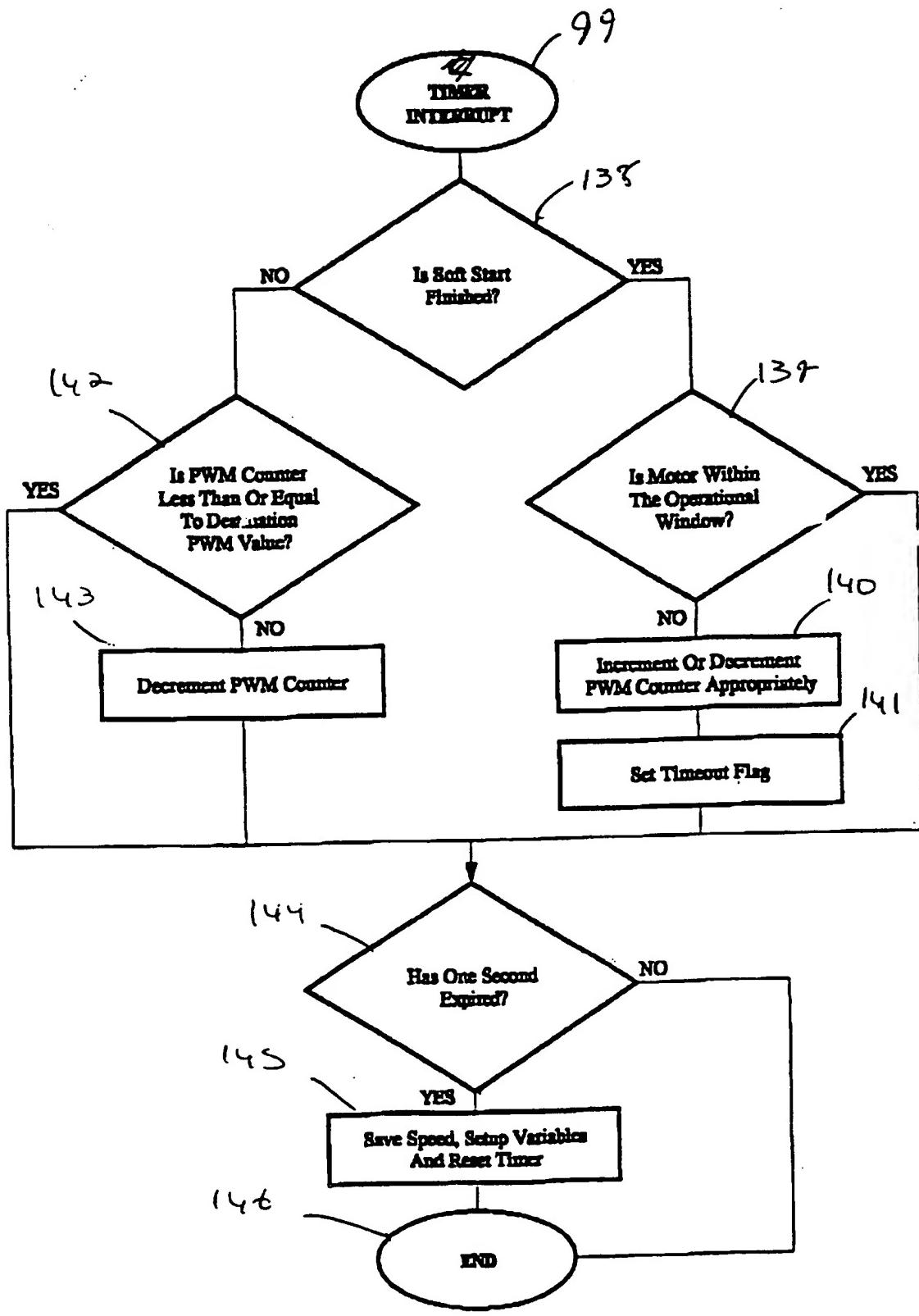
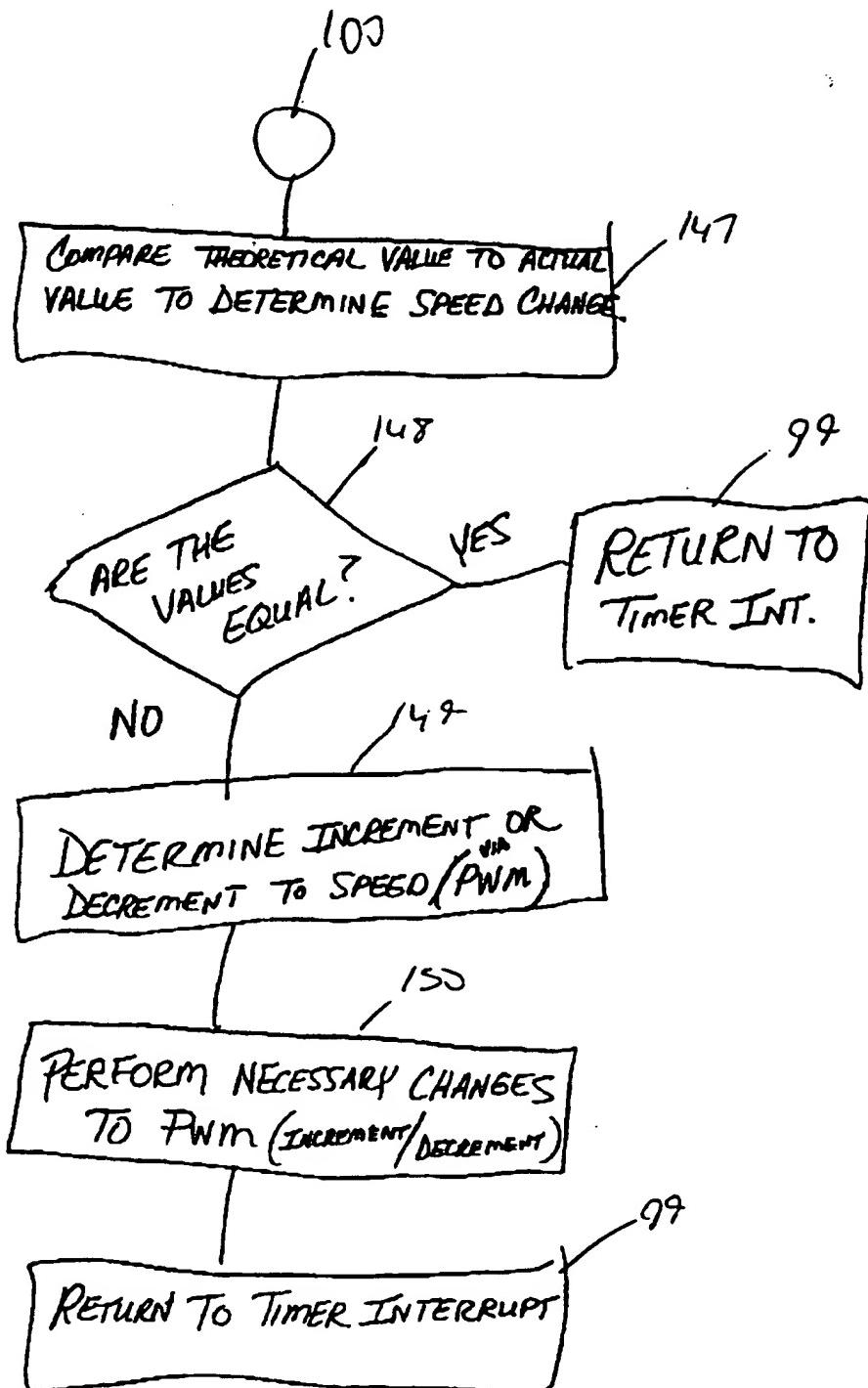


FIG. 22



F16.23

19/28

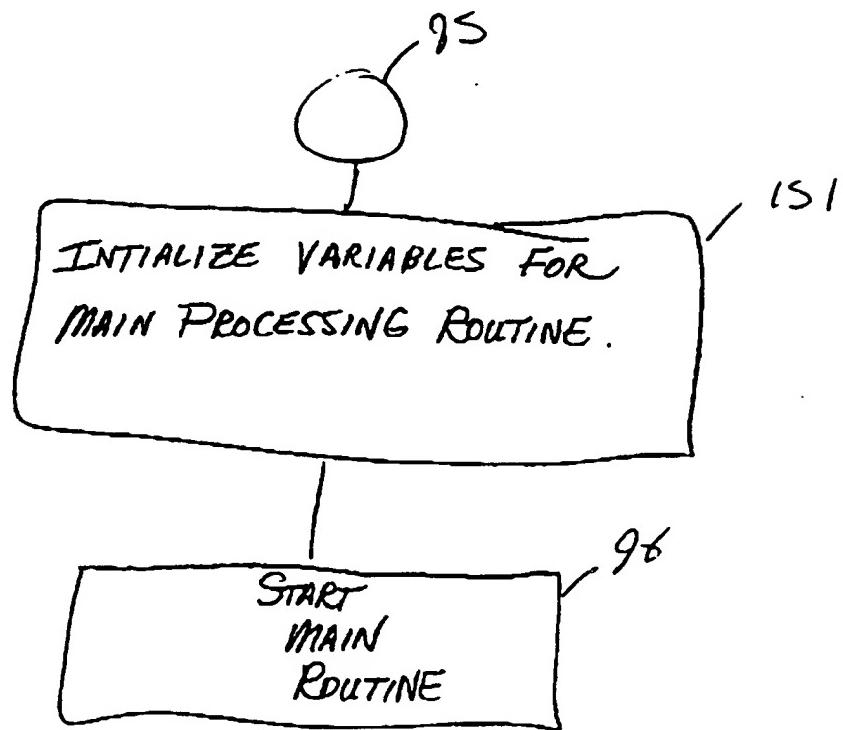
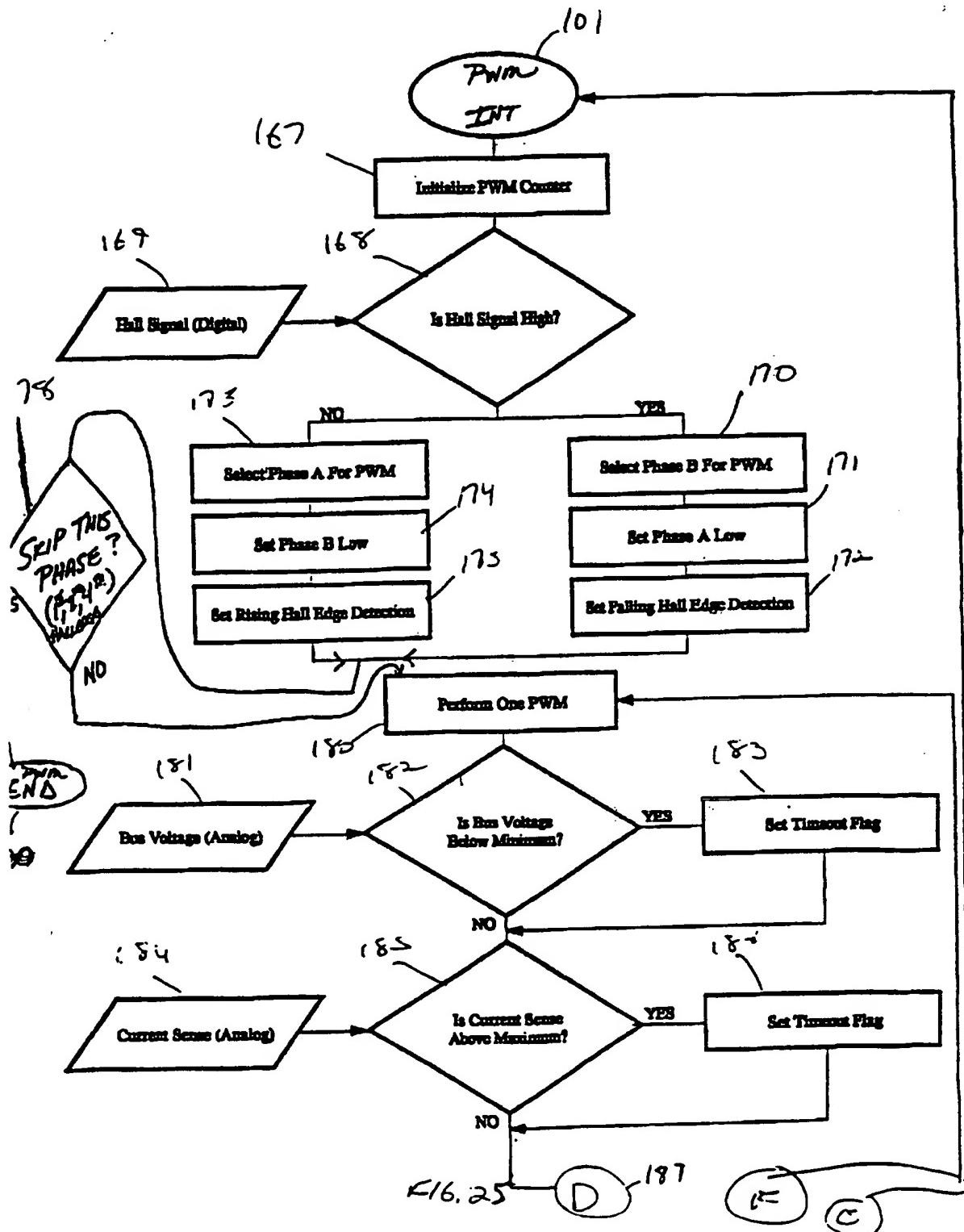


FIG. 24



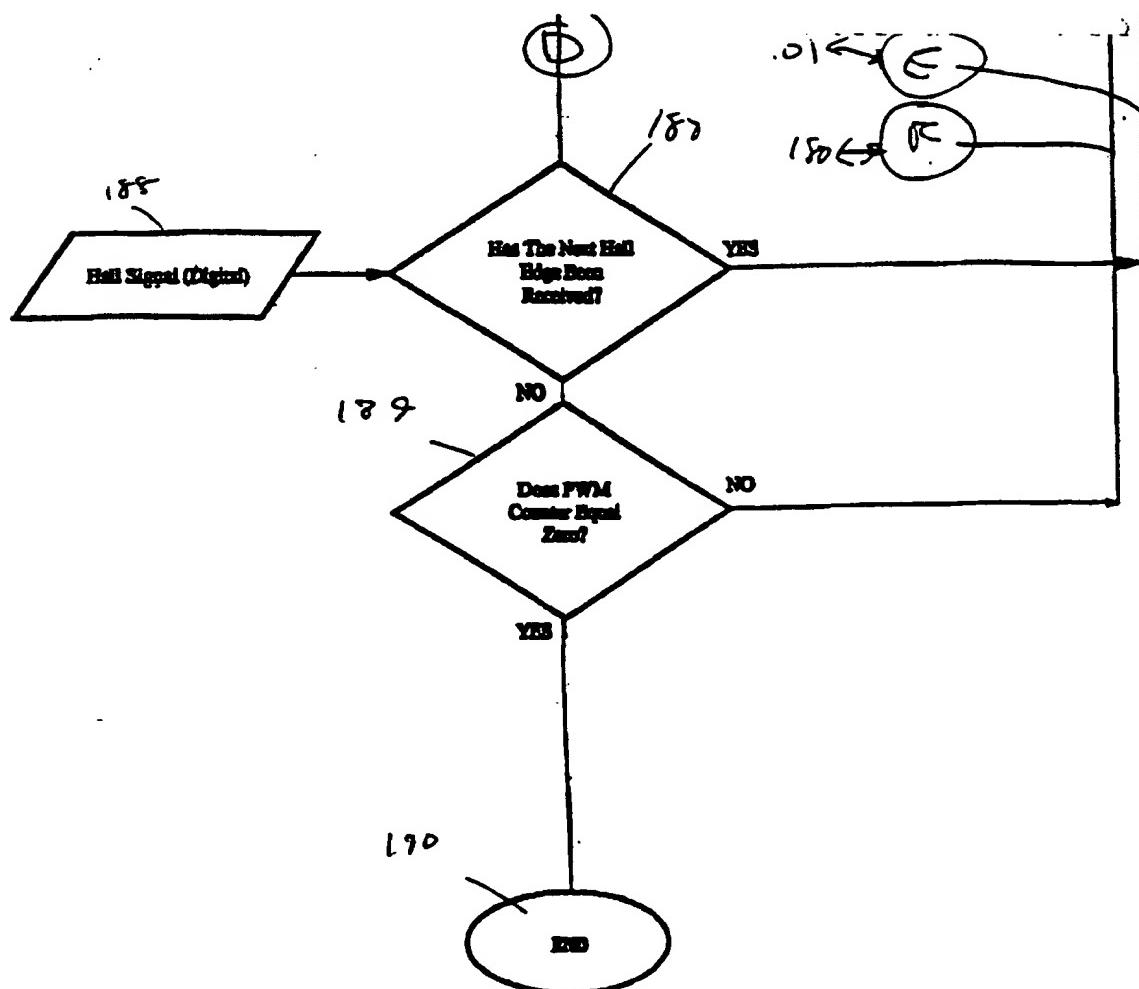


FIG. 26

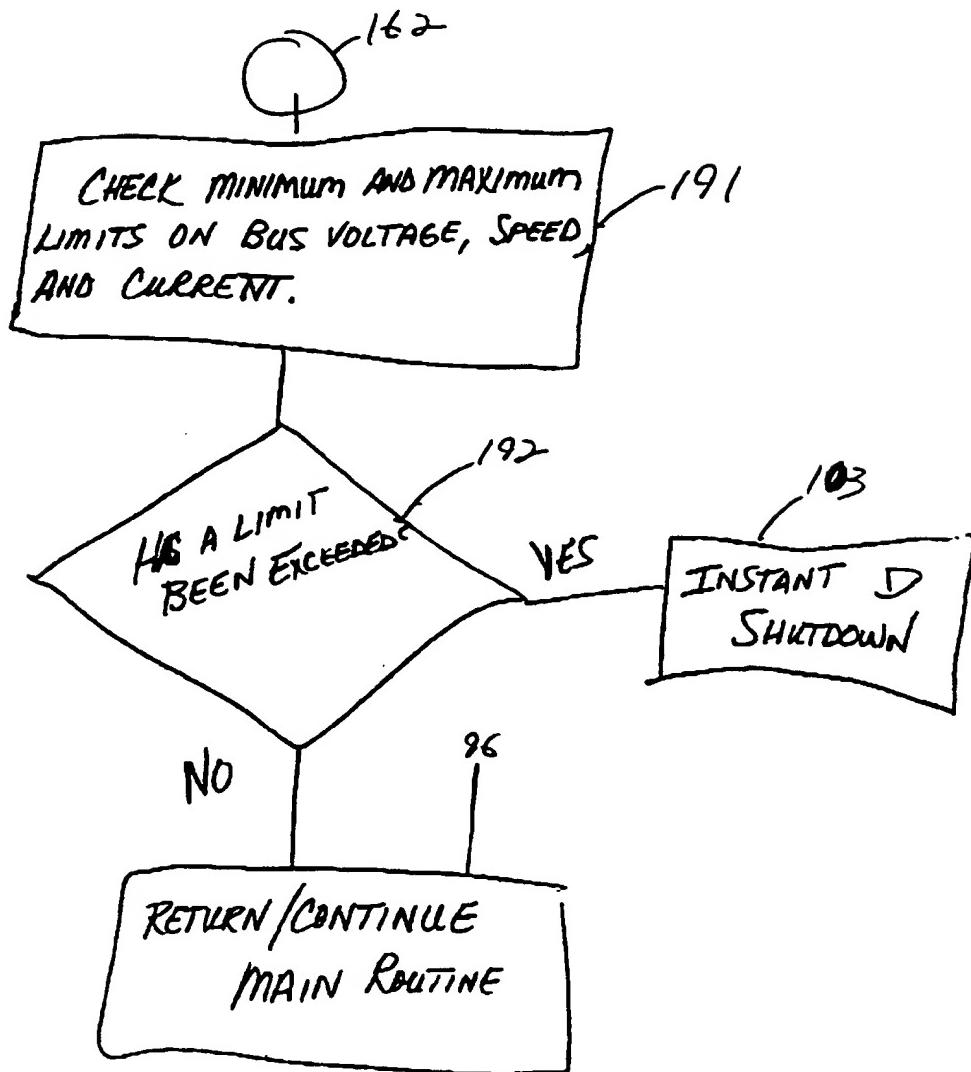


FIG. 27

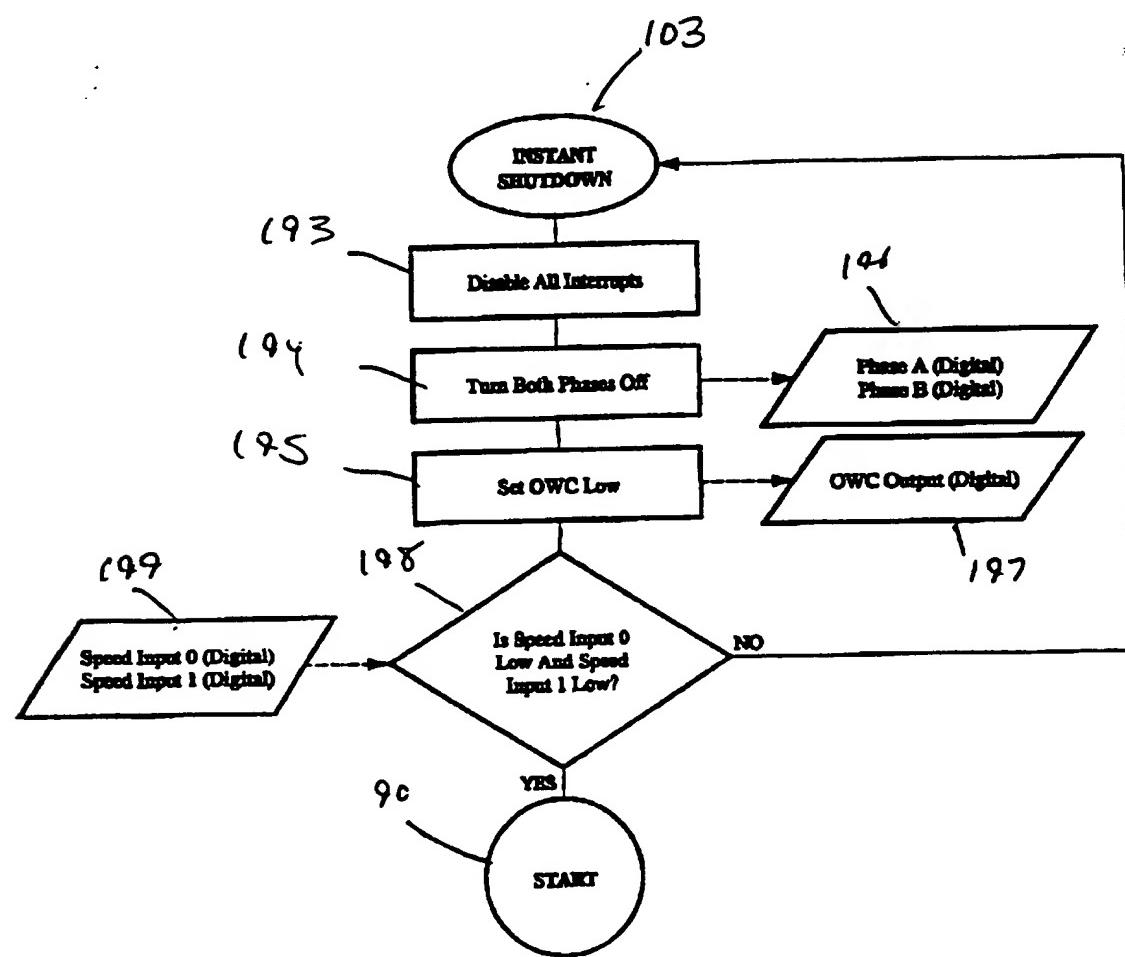


FIG. 28

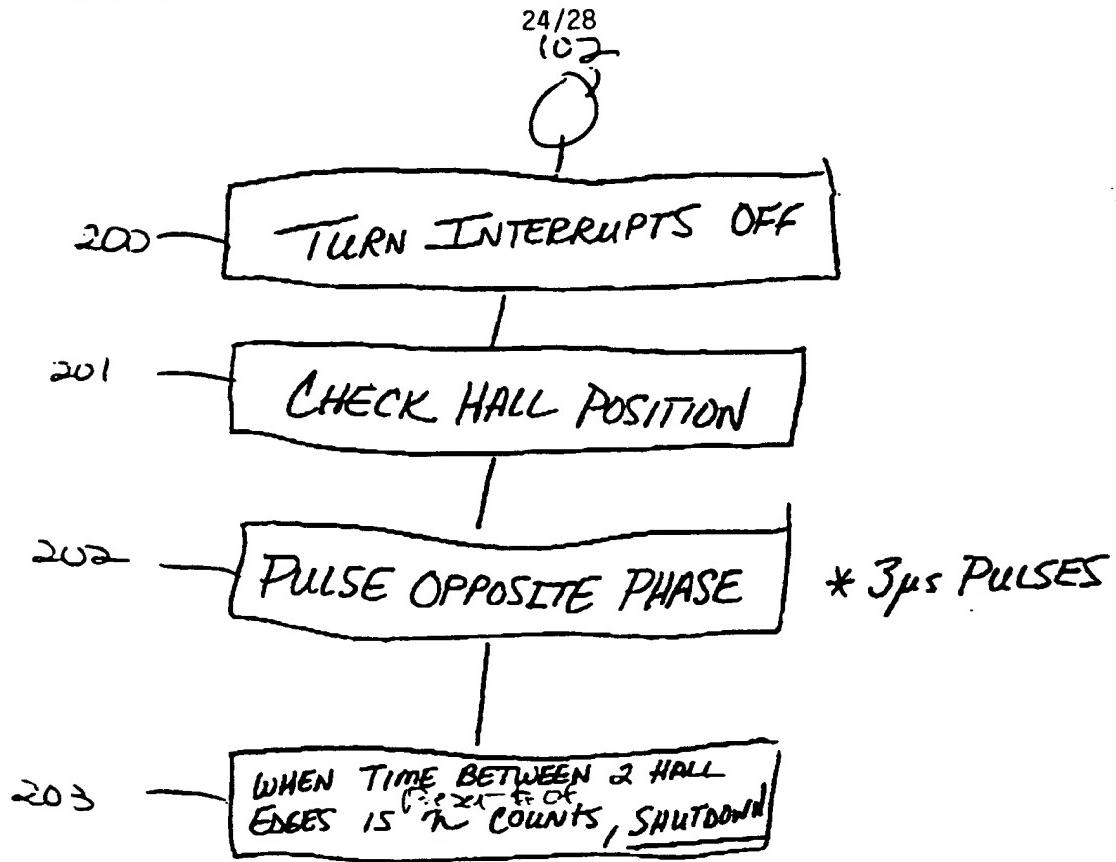
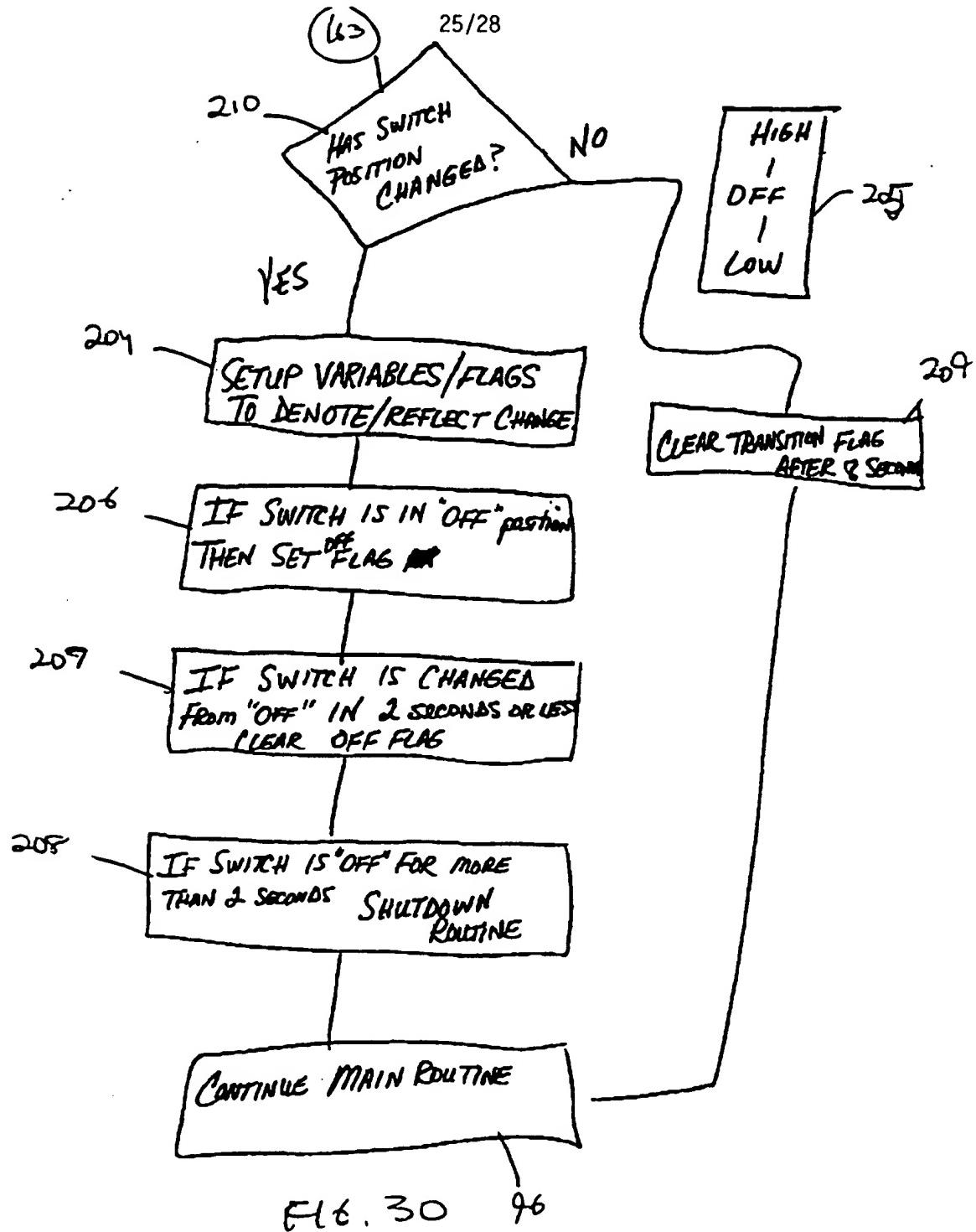


FIG. 29



26/28

164

211

**CHECK PCYCLE VALUE**

212

**BRANCH BASED ON PCYCLE:**

1. SAVE SPEED / SETUP BUS VOLTAGE
2. SAVE SPEED / SETUP CURRENT
3. SAVE BUS VOLTAGE / SETUP SPEED
4. SAVE CURRENT / SETUP SPEED
5. PERFORM CONVERSION

# 5 is performed  
EVERY OTHER  
CYCLE.

213

**INCREMENT PCYCLE**

96

**RETURN TO MAIN**

FIG. 34

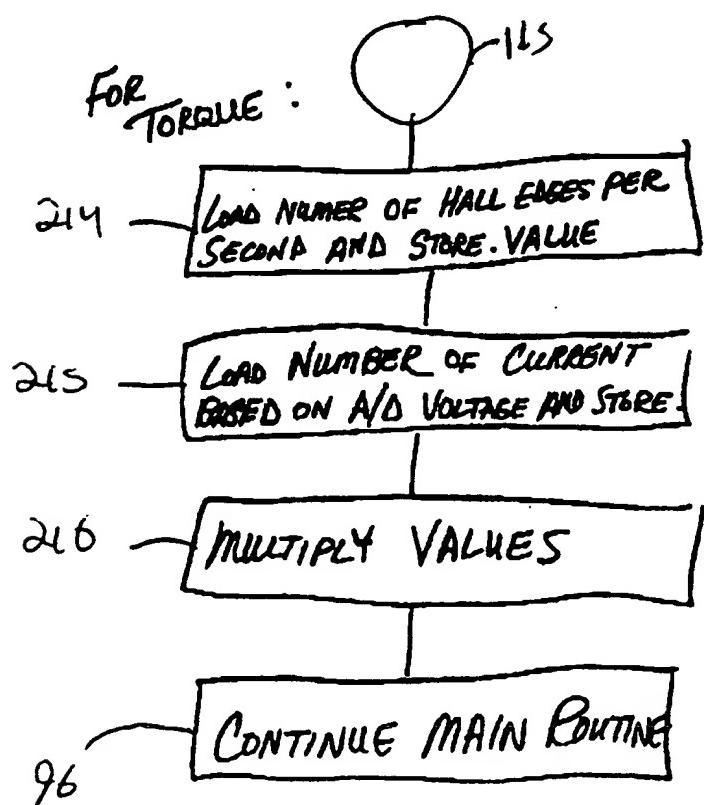


FIG. 35

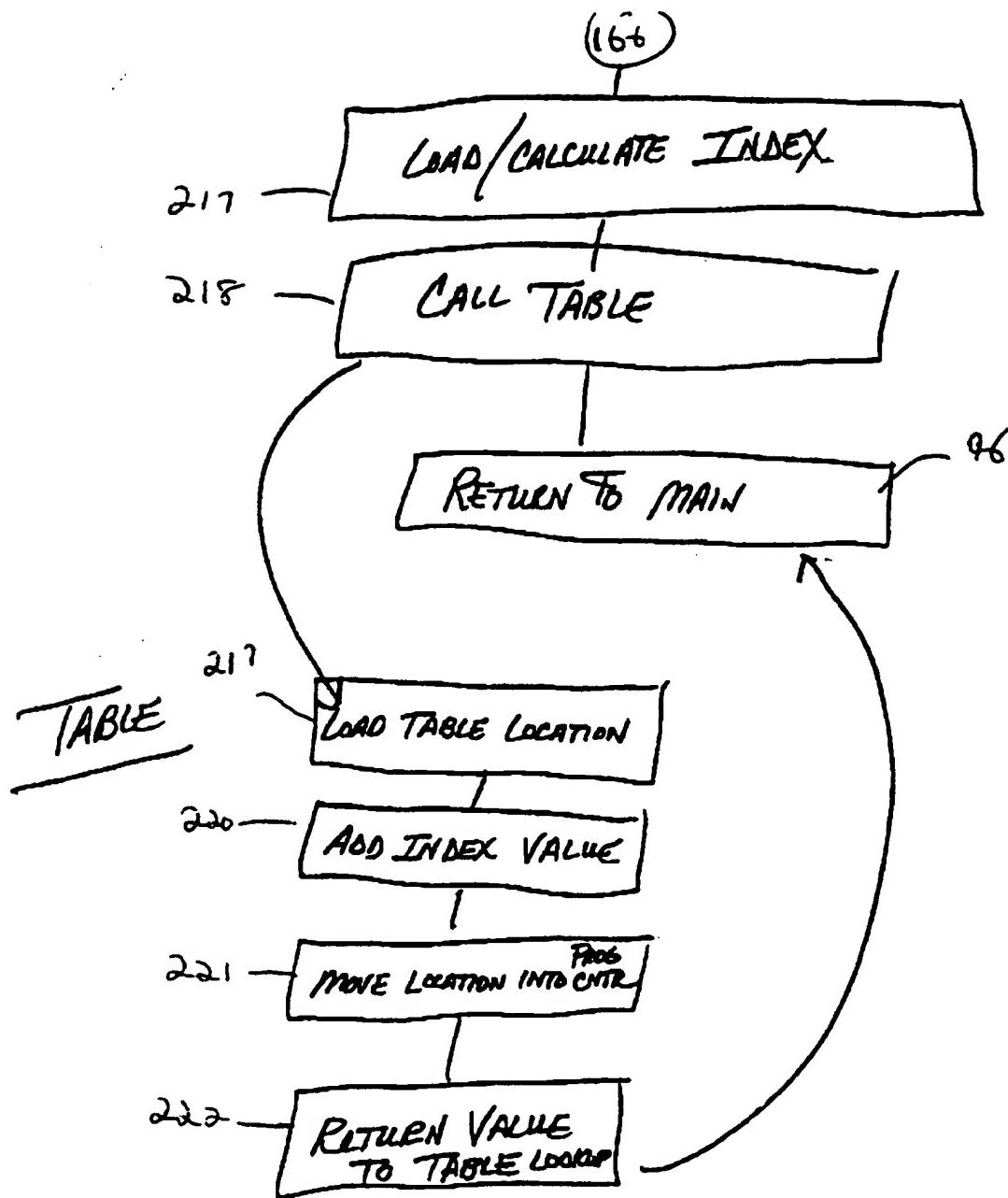


FIG. 36

PCT

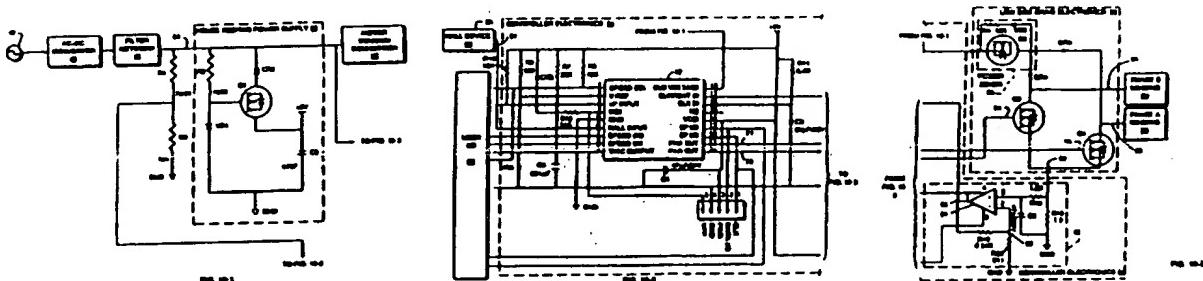
WORLD INTELLECTUAL PROPERTY ORGANIZATION  
International Bureau



INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : <b>H02K 29/00</b>		A3	(11) International Publication Number: <b>WO 97/15111</b>
			(43) International Publication Date: <b>24 April 1997 (24.04.97)</b>
(21) International Application Number: <b>PCT/US96/16728</b>		(81) Designated States: AL, AM, AT, AU, AZ, BB, BG, BR, BY, CA, CH, CN, CZ, DE, DK, EE, ES, FI, GB, GE, HU, IS, JP, KE, KG, KP, KR, KZ, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, TJ, TM, TR, TT, UA, UG, US, UZ, VN, ARIPO patent (KE, LS, MW, SD, SZ, UG), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).	
(22) International Filing Date: <b>17 October 1996 (17.10.96)</b>		Published	
(30) Priority Data: <b>60/005,677 17 October 1995 (17.10.95) US</b>		With international search report. Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.	
(71) Applicant (for all designated States except US): <b>FASCO INDUSTRIES, INC. [US/US]; Suite 200, 500 Chesterfield Center, Chesterfield, MO 63017 (US).</b>		(88) Date of publication of the international search report: <b>5 June 1997 (05.06.97)</b>	
(72) Inventors; and (75) Inventors/Applicants (for US only): <b>GUNNELS, Ronald, Lee [US/US]; 7082 E. Farm Road, #164, Rogersville, MO 65807 (US). SNIDER, Jeffrey, R. [US/US]; 3668 W. Maplewood, Springfield, MO 65807 (US). BUTTRAM, Darrel, C. [US/US]; Route 2, Box 217B, Crane, MO 65633 (US).</b>			
(74) Agents: <b>PERREAUXT, Donald, J. et al.; Lorusso &amp; Loud, 440 Commercial Street, Boston, MA 02109 (US).</b>			

(54) Title: **A BRUSHLESS DC MOTOR ASSEMBLY**



(57) Abstract

A brushless dc motor assembly including a brushless dc motor, and a control board having at least one output connected to a stator winding of said brushless dc motor for providing current flow to said stator winding. Control electronics on the control board control current flow to said stator winding using a microprocessor and based on desired motor operating characteristics. The control electronics include a housekeeping power supply for providing a stable 5v DC signal from a rectified AC line voltage. The control electronics also include a MOSFET output amplifier having a power zener diode connected to the drain thereof, said power zener dissipating temporary back emf resulting from switching of said MOSFET from an on to an off state. A hall device mounted to a stator of said brushless dc motor is also provided. The hall device provides a signal representative of the rotational speed of a rotor of said motor to the control board. The control electronics on said control board control said current flow to said stator winding responsive to said signal from the hall device. The control board is preferably attached to a heatsink whereby the heatsink is attached to MOSFETS on said control board for dissipating heat generated by said MOSFETS.

## INTERNATIONAL SEARCH REPORT

International application No.  
PCT/US96/16728

## A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) : H02K 29/00

US CL : 310/68D

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 310/68D

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

APS

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5,537,015 A [KARWATH] 16 JULY 1996, Col. 5, line 59 -Col. 6, line 31; Col. 4, lines 36-39 and 11-24.	1,2,3
X	US 4,636,936 A [BOYD ET AL.] 13 January 1987, Claim 1	1,5,8,9
X	US 5,410,229 A [SEBASTIAN et al.] 25 April 1995, Col. 6, line 62- Col. 4, line 23.	6
X	US 5,486,747 A [WELCH] 23 January 1996, Claim 1, Col. 1, lines 64-67, Col. 4, line 54-Col. 5, line 14	5, 7-9
X	US 5,331,258 A [LANKIN ET AL.] 19 July, 1994, Fig. 5, Col. 7, lines 1-50	4,5

 Further documents are listed in the continuation of Box C. See patent family annex.

* Special categories of cited documents:	T	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A" document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E" earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O" document referring to an oral disclosure, use, exhibition or other means		
"P" document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search

19 FEBRUARY 1997

Date of mailing of the international search report

24 APR 1997

Name and mailing address of the ISA/US  
Commissioner of Patents and Trademarks  
Box PCT  
Washington, D.C. 20231

Facsimile No. (703) 305-3230

Authorized officer

BURTON S. MULLINS

Telephone No. (703) 305-7063

## INTERNATIONAL SEARCH REPORT

International application No. PCT/US96/16728
---

## C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5,317,245 A [MORITZ ET AL.] 31 May, 1994, Col. 2, line 43; Claim 1	3 ~
X	US 4,626,750 A [POST] 02 December 1986, Col. 5, lines 12-20	4 ✓
Y	US 4,959,797 A [MCINTOSH] 25 September 1990, Abstract, Claim 1.	1-3,5,8-9 ~
Y	US 4,958,269 A [GRITTER] 18 September 1990, Abstract	5,8 ✓
Y	US 4,893,067 A [BHAGWAT ET AL.] 09 January 1990, Abstract	6 ✓
Y	US 4,528,486 A [FLAIG ET AL.] 09 July 1985, Abstract	5